République Tunisienne
Ministère de l'Education

GÉNIE ELECTRIQUE

4ème année de l'enseignement secondaire
Sciences Techniques

Manuel de cours

Les auteurs

Ammar MZOUGHI
Inspecteur Principal des lycées et collèges

Ali ZITOUNI
Inspecteur des lycées et collèges

Hatem LABIDI
Technologue

Mounir BEN HENDA
Professeur Principal hors classe

Fethi AYARI
Professeur Principal

Les évaluateurs

Fredj JAZI
Inspecteur Général de l'éducation

Mohamed BEN HMIDA
Inspecteur Général de l'éducation

Fateh BJAOUI
Technologue

Centre National Pédagogique



Avant-propos

*La **Technologie**, comme discipline d'enseignement, vise l'**acquisition de capacités** générales permettant à l'élève "**d'apprendre à apprendre**". La didactique de cette discipline ne vise pas l'embauche ni la formation à des savoirs strictement professionnels, mais des **méthodes** et des r**aisonnements transférables.***

*Les savoirs constitutifs de la sphère technologique comprennent des connaissances scientifiques, techniques, méthodologiques et des savoir-faire. Leur structuration et la construction de compétences requises par l'état actuel du système productif ne relève **pas** d'un processus d'**accumulation** de solutions techniques mais du développement par l'élève d'**outils conceptuels génératifs,** c'est à dire de connaissances et d'instruments intellectuels de **représentation** et de **traitement des données** lui permettant d'**engendrer lui-même**, face à des **problèmes nouveaux,** des **solutions adéquates**.*

*Cet ouvrage est destiné à des élèves qui préparent le baccalauréat Sciences et Techniques. Ses auteurs ont tenté de créer des points de branchement avec le **monde de l'industrie**. Ils ont présenté au début de chaque thème traité, une situation problème construite autour d'un **système réel** choisi de l'environnement pour créer chez l'élève l'envie d'apprendre. Un problème posé autour du système d'étude constitue à la fois :*

- ✔ ***un critère d'apprentissage**, puisqu'il permet de vérifier, au terme d'une séquence d'enseignement, qu'une notion a été bien assimilée par les élèves. Il s'analyse, dans ce cas, en termes d'**évaluation.***
- ✔ ***un mobile d'apprentissage**, dans la mesure où il suscite l'intérêt de la classe pour le sujet du jour. Il relève dans ce cas de la **motivation.***

*Le problème fonctionne comme le générateur même de l'apprentissage, puisque c'est autour de lui que va se nouer tout le **dispositif didactique**. A chaque problème correspond une **activité élève** montée autour d'un matériel de laboratoire qui peut être un système didactisé, un sous-système ou à défaut une maquette ou un logiciel de simulation. L'élève est alors renvoyé vers un manuel d'accompagnement appelé **"Manuel d'activités"** qui lui servira de guide de recherche et de découverte des nouveaux **concepts** et des nouvelles **méthodes** de résolution.*

*Une page de **résumé** à la fin de chaque séquence recense l'essentiel. La situation de départ est reprise comme support de **validation** des apprentissages, suivie d'un ensemble de **situations similaires intégratives** sur lesquelles l'élève applique ses nouveaux acquis et vérifie son niveau d'assimilation.*

*Les auteurs espèrent que ce manuel contribue à aider l'**enseignant** à **préparer** ses séquences d'enseignement et l'**élève** à mieux **assimiler** les concepts et les méthodes.*

Les auteurs

Sommaire

		Pages
Avant-propos		3
Sommaire		4
PARTIE AUTOMATIQUE		
Chapitre A1 : Opérations arithmétiques binaires		
	Leçon A1-1 : Addition et Multiplication en binaire	6
	Leçon A1-2 : Soustraction et Division en binaire	31
Chapitre A2 : Logique combinatoire		
	Leçon A2-1 : Les comparateurs logiques	40
	Leçon A2-2 : Unité arithmétique et logique	46
Chapitre A3 - Logique séquentielle		
	Leçon A3 : Les compteurs	57
Chapitre A4 - Logique programmée		
	Leçon A4-1 : Automate Programmable Industriel (API)	66
	Leçon A4-2 : Microcontrôleurs	97
Chapitre A5 Notions d'asservissement linéaire		
	Leçon A5 : Asservissement linéaire	132
PARTIE ELECTROTECHNIQUE		
Chapitre B1 : Circuit électrique en monophasé		
	Leçon B1-1 : Circuits électriques en monophasé	148
	Leçon B1-2 : Système triphasé	171
Chapitre B2 : Transformateur monophasé		
	Leçon B2 : Les transformateurs monophasés	183
Chapitre B3 : Moteur asynchrone		
	Leçon B3 : Les moteurs asynchrones triphasés	191
Chapitre B4 : Moteur à courant continu		
	Leçon B4 : Les moteurs à courant continu	207
PARTIE ELECTRONIQUE		
Chapitre C1 : Les amplificateurs linéaires intégrés		
	Leçon C1 : Les amplificateurs linéaires intégrés	225
Bibliographie		240

Automatique

A1 Opérations arithmétiques binaires

A2 Logique combinatoire

A3 Logique séquentielle

A4 Logique programmée

A5 Notions d'asservissement linéaire

Partie Automatique

L'ADDITION ET LA MULTIPLICATION EN BINAIRE

A- MISE EN SITUATION

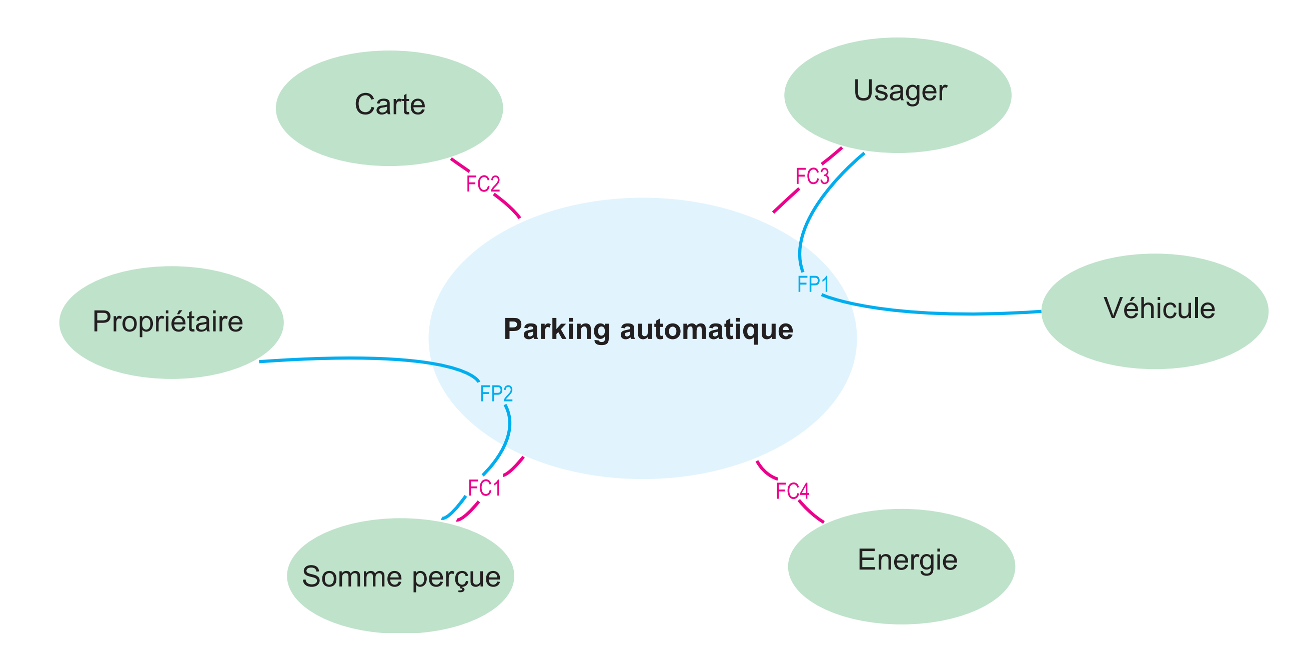
1- Fonctionnement du système : gestion d'un parking

Un parking public payant à double issues, débouche sur 2 rues opposées, et est accessible aux usagers suivant deux modes :

1- Mode abonné : dans ce cas l'usager doit introduire sa carte d'abonnement dans un lecteur de cartes installé à l'entrée du parking. L'accès n'est autorisé que si la carte est valide.

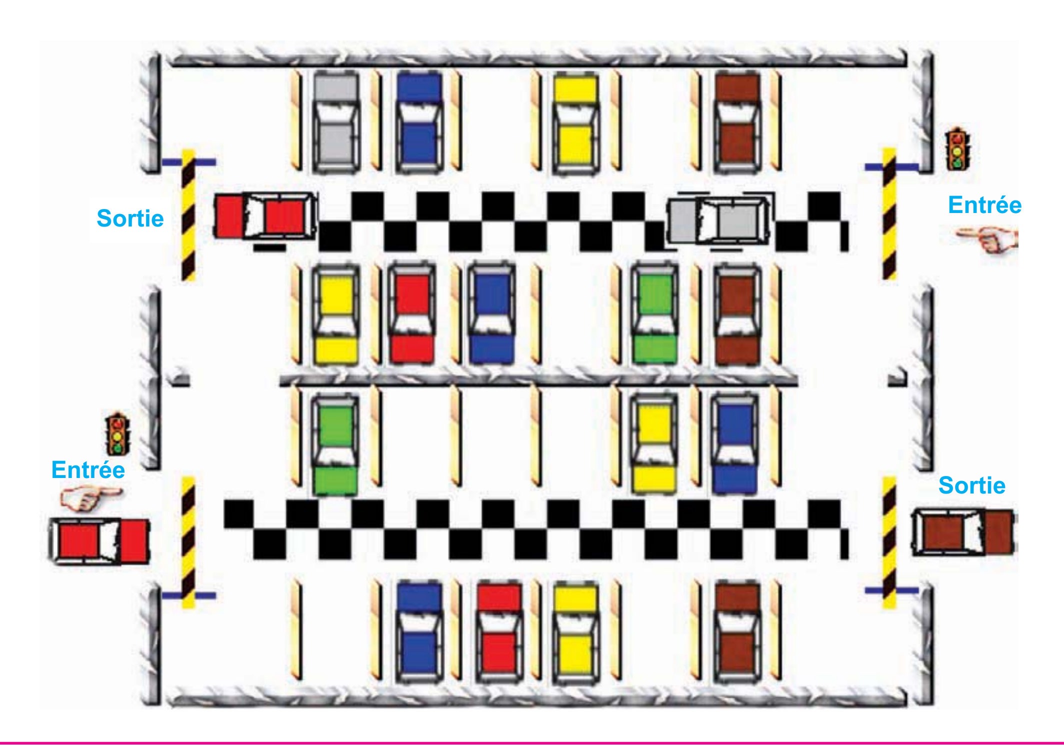


2- Mode passager : dans ce cas pour pouvoir y accéder, l'usager est appelé à s'acquitter d'une somme forfaitaire qu'il doit jeter dans un panier (somme fixée d'avance à un dinar cinq cents millimes en pièces de 500mil et / ou de 1 dinar uniquement).
Une signalisation lumineuse installée à chacune des entrées, affiche complet lorsque les places disponibles sont épuisées.
Dans les deux cas cités précédemment, si l'usager insiste alors qu'il est dans une situation illégale, une signalisation sonore et une autre lumineuse se mettent en marche.
Recensons quelques-unes des fonctions que doit remplir ce système :



Les expressions des fonctions de service inventoriées précédemment :

- ➪ **FP1** : Permettre à l'usager de garer son véhicule sans risque de contravention.
- ➪ **FP2** : Permettre au propriétaire de gérer les sommes payées par les usagers.
- ➪ **FC1** : Evaluer la somme payée par l'usager.
- ➪ **FC2** : Valider le code client.
- ➪ **FC3** : Signaler à l'usager l'épuisement des places.
- ➪ **FC4** : Le système doit fonctionner à l'énergie électrique.



Problématique : Comment peut-on gérer ce parking ?

Pour satisfaire chacune des fonctions de service citées précédemment, il faut la traduire en solution technique.
A titre d'exemple, prenons le cas de **FC3** et **FP2**.
Pour **FC3** le système doit être capable de totaliser le nombre des véhicules entrants, retrancher le nombre de véhicules sortants et avertir les usagers lorsque les places disponibles sont épuisées. D'où deux opérations à réaliser : une addition et une soustraction.
Pour **FP2** par exemple il s'agit de faire le total des sommes payées par les usagers durant une journée d'ouverture du parking, cette fonction nécessite une opération d'addition ou / et une opération de multiplication, etc.....

Après analyse de ces fonctions, on peut déduire que la gestion de ce parking nécessite des structures électroniques pouvant réaliser : des additions, des soustractions, des multiplications, des comparaisons, etc... en plus des opérations d'affichage et de signalisation.

B- LES OPERATIONS BINAIRES DE BASE

Généralement les opérations de base que l'on est amené à mettre en oeuvre dans des systèmes numériques de ce type, sont souvent :
La **somme**, ou **addition** et l'opération inverse qui est la **différence** ou **soustraction**.
Le **produit**, ou **multiplication** et l'opération inverse qui est le **quotient** ou **division**.
L'**élévation** à une **puissance** $\mathbf{n}^{ième}$ et son opération inverse, l'extraction de la **racine** $\mathbf{n}^{ième}$.
Le contenu proposé, traitera uniquement des quatre premières opérations, à savoir l'**addition**, la **soustraction**, la **multiplication** et la **division**.
Pour distinguer ces opérations, on utilise comme convenu les symboles classiques relatifs à l'algèbre classique, à savoir (le **+** pour *l'addition*, le **-** pour la *soustraction*, le **X** pour la *multiplication* et le **/** pour la *division*).
La mise en œuvre des différentes opérations est régie par des règles strictes, qui font l'objet de l'arithmétique binaire.

NB : Ces différentes opérations sont susceptibles d'être appliquées à des nombres positifs et négatifs; d'où la nécessité d'une convention pour la représentation de ces nombres.

Ecriture des nombres signés :
- Notation signe-grandeur : dans ce cas le nombre est représenté par un groupe de bits appelé bits de grandeur ou norme, représentant l'équivalent binaire exact de la valeur décimale devancé par un bit de signe. Ce bit de signe prend la valeur zéro pour dire que le nombre est positif et 1 pour dire que le nombre est négatif.
Exemple : **(+10)** ==> **0** 1010 donc **(-10)** ==> **1** 1010
- Notation en complément à 2 : s'obtient en inversant les bits formant l'équivalent binaire du nombre et en ajoutant 1 au terme du poids le plus faible.
L'écriture des nombres signés dans ce cas de figure est comme suit :
- Si le nombre est positif : la grandeur est la grandeur binaire exacte et le bit de signe est un 0.
- Si le nombre est négatif : la grandeur est le complément à 2 de la grandeur exacte et le bit de signe est un 1.

Remarque : Cette partie sera expliquée plus en détail lors de l'étude de l'opération de soustraction.

1- L'addition binaire :

Reprenons la fonction **FC1** qui consiste à évaluer la somme d'argent payée par l'usager. Cette fonction est accomplie en deux étapes, une première étape consiste à totaliser la somme des pièces jetées par l'usager d'où une opération d'**addition**, et la deuxième étape, consiste à valider cette somme, d'où une opération de **comparaison.**

Problème technique : Quelle est la structure électronique de base qui permet d'effectuer une opération d'addition de deux nombres binaires ?

Activité
Réaliser l'activité **N°1** du **TP A1-1** dans le manuel d'activités : cette activité consiste à chercher la somme de deux nombres binaires à un bit chacun (**A :** a_0) et (**B :** b_0), la retenue qui peut en découler suite à cette opération et par la suite dégager la ou les limites de ce circuit.

1-1 Principe :

Le principe est le même qu'en décimal ; en règle générale, pour l'addition, on inscrit le premier terme, puis en dessous, le ou les termes suivants, en plaçant dans les mêmes colonnes les chiffres de poids identiques.

La numération étant dans la base **2**, tout résultat supérieur à **1** génère un report dans la colonne suivante (par analogie à la base 10 dans laquelle tout résultat supérieur à 9, génère une retenue à reporter dans la colonne suivante).

Rappelons par un exemple les principes de l'addition en décimal : soit à additionner les deux nombres suivants : **67** et **43**.

Pondération	10^2	10^1	10^0	
Reports	1	1		
		6	7	1er terme
+		4	3	2ème terme
	1	**1**	**0**	**Résultat**

Appliquons ces principes sur les mêmes valeurs, mais en binaire.

La conversion en binaire des deux opérandes donne les résultats ci-contre :

Décimal		Binaire
67	⇒	**1000011**
43	⇒	**101011**

Sachant aussi que la table d'addition en binaire est comme suit :

a_0	b_0	r	S
0	0	0	0
0	1	0	1
1	1	1	0
1	0	0	1

NB : dans le cas où on sera amené à additionner 3 uns comme suit : **1 + 1 + 1** le résultat final donnera un **1** pour **S** et un **1** pour **r**.

D'où le résultat de l'addition :

Pondération	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Reports					1	1		
	1	0	0	0	0	1	1	1er terme
+		1	0	1	0	1	1	2ème terme
	1	**1**	**0**	**1**	**1**	**1**	**0**	**Résultat**

Ainsi, on obtient :

$1101110_{(2)}$ ==> **110** $_{(10)}$

1-2 Réalisation industrielle :

a/ Demi-additionneur :

Pour une addition de deux nombres **A (A = a_0)** et **B (B = b_0)** de **1 bit**, **4** combinaisons sont possibles, et le résultat occupera **2 bits**, un bit pour la somme (**S**) et un autre pour la retenue (**r**). Ce dispositif est également appelé demi-additionneur ou en anglais (**Half-adder**)

Sa table de vérité est la suivante :

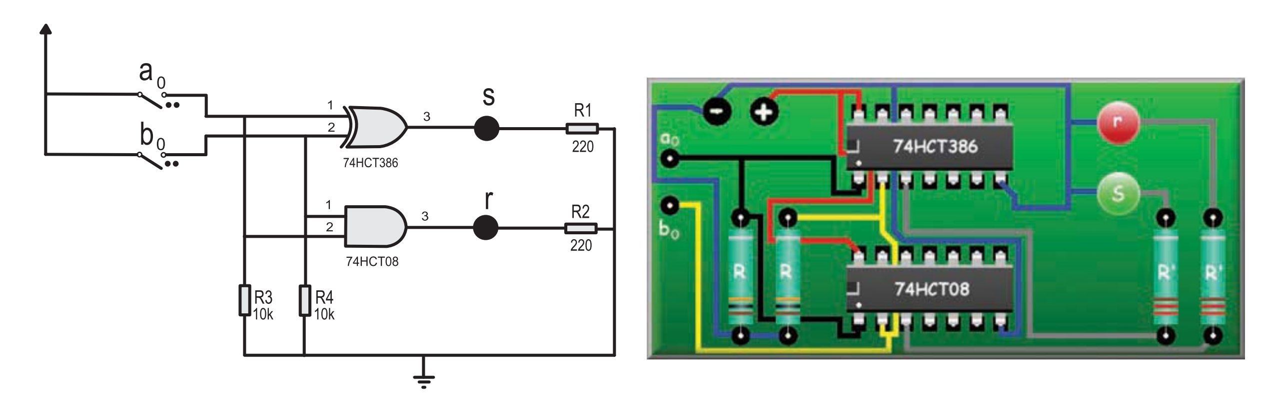
a_0	b_0	r	S
0	0	0	0
0	1	0	1
1	1	1	0
1	0	0	1

Ses expressions logiques sont donc :

$$S = \overline{a_0}b_0 + a_0.\overline{b_0} = a_0 \oplus b_0$$

$$r = a_0.b_0$$

D'où le logigramme correspondant ainsi qu'un exemple de cablage :



b/ Additionneur complet de deux nombres à 1 bit :

Le **demi-additionneur** étudié précédemment ne permet pas en réalité de réaliser totalement une addition.

Considérons à cet effet, l'addition de deux nombres à plusieurs bits. On s'aperçoit que dans certains cas, il est nécessaire, non pas d'additionner **2 bits** de même poids, mais **3 bits**, le troisième bit provenant d'une retenue résultante de l'addition des bits du rang précédent.

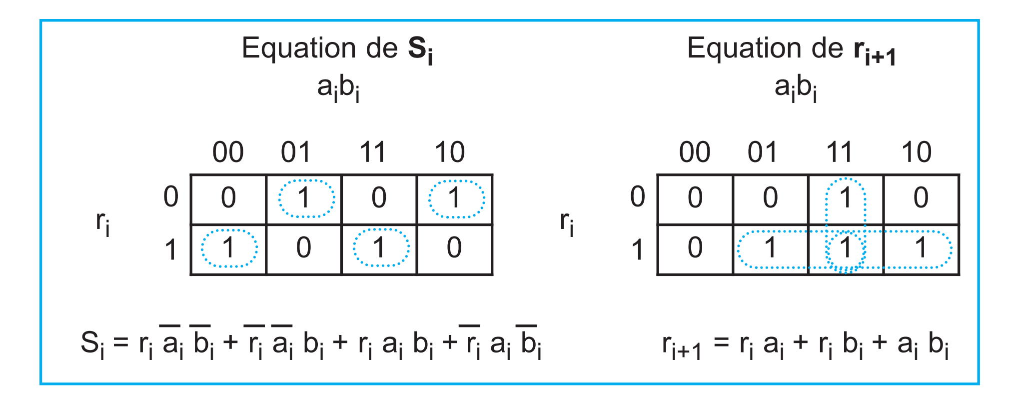
Activité

Réaliser l'activité **N°2** du **TP A1-1** dans le manuel d'activités : Cette activité consiste à remédier au défaut rencontré lors de l'étude du **demi-additionneur.**

Reproduisons la table de vérité de l'additionneur complet obtenue lors de l'activité **N°2 :**

a_i	b_i	r_i	r_{i+1}	S_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Les expressions logiques sont les suivantes :



Equation de $\mathbf{S_i}$

r_i \ a_ib_i	00	01	11	10
0	0	1	0	1
1	1	0	1	0

$S_i = r_i\,\overline{a_i}\,\overline{b_i} + \overline{r_i}\,\overline{a_i}\,b_i + r_i\,a_i\,b_i + \overline{r_i}\,a_i\,\overline{b_i}$

Equation de $\mathbf{r_{i+1}}$

r_i \ a_ib_i	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$r_{i+1} = r_i\,a_i + r_i\,b_i + a_i\,b_i$

Après développement, on peut écrire l'équation de $\mathbf{S_i}$ sous cette forme :

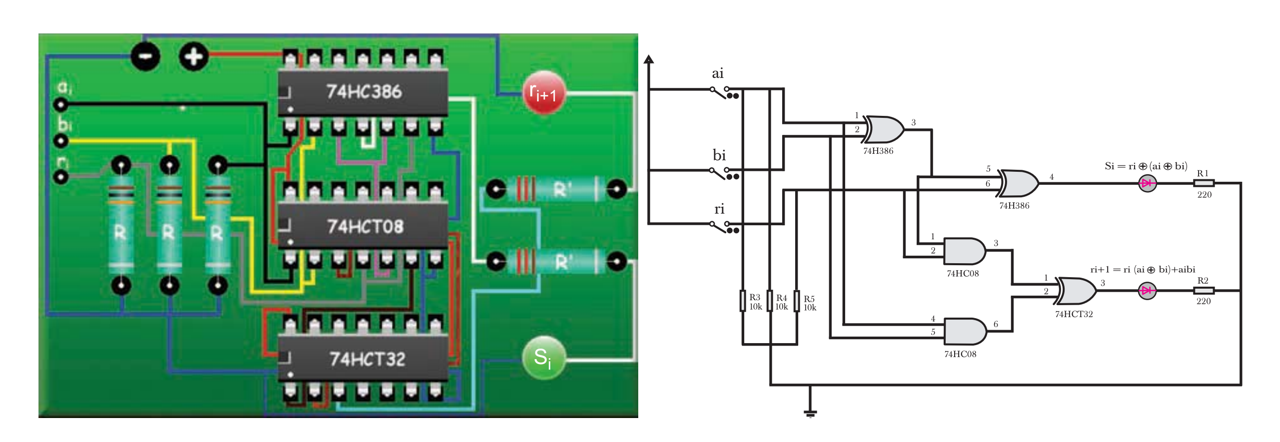
$$S_i = r_i\,\overline{a_i}\,\overline{b_i} + \overline{r_i}\,\overline{a_i}\,b_i + r_i\,a_i\,b_i + \overline{r_i}\,a_i\,\overline{b_i}$$
$$= r_i\,(\overline{a_i}\,\overline{b_i} + a_i\,b_i) + \overline{r_i}\,(\overline{a_i}\,b_i + a_i\,\overline{b_i})$$
$$= r_i\,\overline{(a_i \oplus b_i)} + \overline{r_i}\,(a_i \oplus b_i) = r_i \oplus (a_i \oplus b_i)$$

De même pour l'équation de r_{i+1}

$$r_{i+1} = r_i\,\overline{a_i}\,b_i + r_i\,a_i\,\overline{b_i} + a_i\,b_i \quad \text{ou} \quad r_i\,a_i + r_i\,b_i + a_i\,b_i$$
$$= r_i\left(\overline{a_i}\,b_i + a_i\,\overline{b_i}\right) + a_i\,b_i$$
$$= r_i\left(a_i \oplus b_i\right) + a_i\,b_i$$

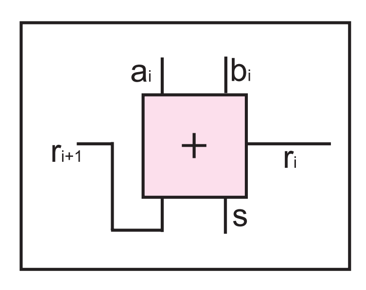
r_i \ a_ib_i	00	01	11	10
0	0	0	1	0
1	0	1	1	1

D'où le logigramme correspondant et un exemple de cablage :





Ainsi que le symbole :

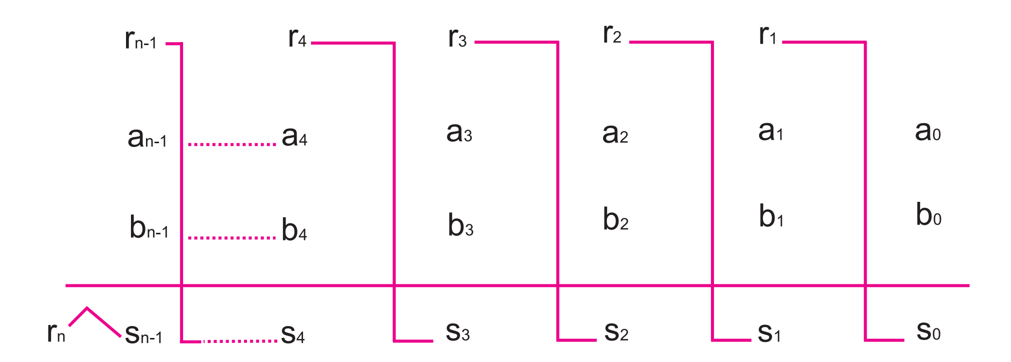


1-3 Généralisation à n bits :

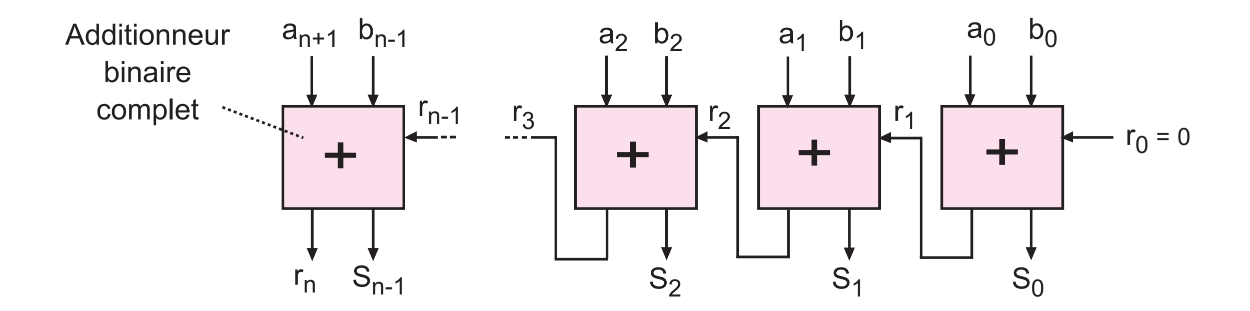
Le principe étant le même, en généralisant, nous pouvons construire un additionneur de deux mots à n bits.

Exemple : soit **$A = a_n.......a_3.a_2.a_1.a_0$** **$B = b_n.......b_3.b_2.b_1.b_0$**

Le principe de l'addition est comme suit :



D'où le logigramme correspondant à base d'additionneur complet de deux nombres à **1 bit.**



Activité

Réaliser l'activité **N°3** du **TP A1-1** dans le manuel d'activités :

L'activité consiste à réaliser un additionneur de deux nombres à 3 bits (**A :$a_2a_1a_0$**), (**B :$b_2b_1b_0$**) en utilisant des additionneurs élémentaires complets de deux nombres à 1 bit chacun branchés en cascade.



1-4 L'addition en complément à 2 :

a/ Cas de deux nombres positifs : l'addition dans ce cas est immédiate

Exemple : soit à additionner (+9) et (+4).

$(+9)_{(10)}$ ==> **0** 1001
$(+4)_{(10)}$ ==> **0** 0100
0 1101 ==> $(13)_{(10)}$

	0	1	0	0	1
+	0	0	1	0	0
	0	1	1	0	1

b/ Cas d'un nombre positif et d'un nombre négatif de module plus petit :

Exemple : soit à additionner (+9) et (-4).

Dans ce cas (+9) sera remplacé par son équivalent binaire exacte; par contre (-4) doit être remplacé par son complément à 2,

d'où :

$(+9)_{(10)}$ ==> **0** 1001
$(+4)_{(10)}$ ==> **0** 0100
Complément à 1 de (+4) = **1** 1011 et donc
le complément à 2 = **1** 1100

	1	1					
+		0	1	0	0	1	$+9_{(10)}$
		1	1	1	0	0	$-4_{(10)}$
	1	0	0	1	0	1	$+5_{(10)}$

c/ Cas d'un nombre positif et d'un nombre négatif de module plus grand :

Exemple : soit à additionner (-9) et (+4).

Dans ce cas (-9) doit être exprimé par son complément à 2, par contre (+4) est replacé par son équivalent binaire exacte,

d'où :

$(+4)_{(10)}$ ==> **0** 0100
$(+9)_{(10)}$ ==> **0** 1001
Complément à 1 de (+9) = **1** 0110 et donc

		1			
+	1	0	1	1	1
	0	0	1	0	0
	1	1	0	1	1

le complément à 2 = **1** 0111. Puisque le bit de signe est 1, le résultat est supposé négatif, dans ce cas la norme du résultat représente le complément à 2 de la valeur recherchée; donc pour obtenir la valeur exacte de l'opération, il faut complémenter à 2 cette norme; d'où : 1011 ==> complément à 1 = 0100 ==> complément à 2 = 0101, c'est bien $5_{(10)}$ et puisque le bit de signe est 1 le résultat réel est (-5).

d/ Cas de deux nombres négatifs :

Exemple : soit à additionner (-9) et (-4).

Dans ce cas les deux nombres doivent être exprimés dans la notation en complément à 2.

(-9) ==> **1** 0111 et (-4) ==> **1** 1100

Puisque le bit de signe est 1, le résultat est négatif; la norme du résultat trouvé n'est autre que le complément à 2 de la valeur exacte. Pour l'obtenir il faut recomplémenter à 2 la norme du résultat trouvé; d'où

	1	1	1			
+		1	0	1	1	1
		1	1	1	0	0
	1	1	0	0	1	1

0011 ==> un complément à 2 = 1101c'est l'équivalent binaire de $13_{(10)}$
et puisque le bit de signe vaut 1, le résultat réel est $-13_{(10)}$

e/ Cas de deux nombres égaux et opposés :

Dans ce cas le résultat est évidement nul.

1-5 Avantage(s) et inconvénient(s) :

Ce type de structure présente l'avantage d'être simple à mettre en œuvre, facile à comprendre, en contre partie, il présente l'inconvénient majeur d'avoir un temps de réponse relativement long à cause de la propagation de la retenue d'un bloc à l'autre, qui suivant les cas, peut aboutir à des résultats erronés, d'où l'appellation de ce type d'additionneur : d'**additionneur** à **propagation** de **retenues** ou **additionneur itératif**.

Pour faire face à cet inconvénient et pour effectuer la somme plus rapidement, il faut éliminer ou au moins limiter ce temps de propagation.

Pour cela on se base sur l'hypothèse que les termes de la somme sont connus et disponibles avant même que commence l'opération d'addition. On peut alors calculer, **en anticipant**, la retenue pour chaque étage indépendamment des étages précédents. Il s'agit de pouvoir disposer de toutes les retenues simultanément et dans un temps le plus court possible.

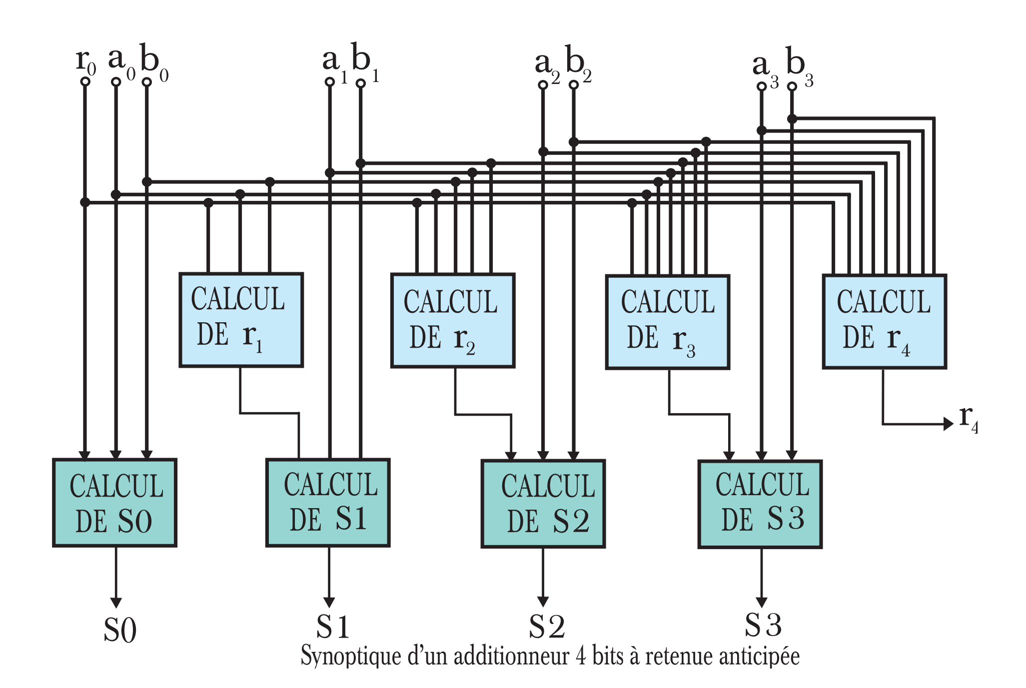
Autrement dit, il faut calculer la retenue r_1 à partir des bits a_0, b_0 et r_0, la retenue r_2 à partir des bits a_0, b_0, r_0, a_1 et b_1 et ainsi de suite.

Ce type d'additionneur est appelé : **additionneur à anticipation de retenues** ou à **report anticipé**.

1-6 Additionneur à anticipation de retenues :

a/ Principe :

Le synoptique suivant, explique le principe de fonctionnement de ce type de circuit :



Synoptique d'un additionneur 4 bits à retenue anticipée

Pour effectuer le calcul des retenues de façon anticipée, il faut transformer l'équation de la retenue r_{i+1} vue précédemment (page 11).

$$r_{i+1} = a_i \overline{b_i} r_i + a_i b_i + \overline{a_i} b_i r_i$$

Cette transformation s'effectuera en se basant sur la constatation suivante :

$$r_{i+1} = 1 \text{ lorsque } a_i = b_i = r_i = 1$$

On peut alors ajouter à l'expression de r_{i+1} la quantité $a_ib_ir_i$ autant de fois que l'on veut sans changer le résultat final.
Ajoutons alors deux fois la quantité $a_ib_ir_i$ à l'expression de r_{i+1}, cette expression devient :

$$r_{i+1} = a_i\,\overline{b_i}\,r_i + a_i\,b_i\,r_i + a_i\,b_i + \overline{a_i}\,b_i\,r_i + a_i\,b_i\,r_i$$

Dans cette expression, mettons en facteur d'une part a_ir_i et d'autre part b_ir_i , l'expression de r_{i+1} devient :

$$r_{i+1} = a_i\,r_i\left(\overline{b_i} + b_i\right) + a_i\,b_i + b_i\,r_i\left(\overline{a_i} + a_i\right)$$

Soit :

$$r_{i+1} = a_i\,r_i + a_i\,b_i + b_i\,r_i$$
$$= a_i\,b_i + r_i\left(a_i + b_i\right)$$

Posons le produit $a_ib_i = g_i$

et la somme $(a_i + b_i) = p_i$

L'expression de r_{i+1}devient alors : $r_{i+1} = g_i + r_ip_i$

D'où les expressions des différentes retenues :

➩ $r_1 = g_0 + r_0p_0$

➩ $r_2 = g_1 + r_1p_1$

Remplaçons r_1 par sa valeur, dans l'expression de r_2

➩ $r_2 = g_1 + (g_0 + r_0p_0)p_1$

Après développement, l'expression de r_2 devient :

➩ $r_2 = g_1 + g_0p_1 + r_0p_0p_1$

De même pour r_3

➩ $r_3 = g_2 + r_2p_2$

Remplaçons r_2 par sa valeur, dans l'expression de r_3 l'expression de r_3 devient :

➩ $r_3 = g_2 + (g_1 + g_0p_1 + r_0p_0p_1)p_2$

D'où :

➩ $r_3 = g_2 + g_1p_2 + g_0p_1p_2 + r_0p_0p_1p_2$

L'application de la même procédure donne :

➩ $r_4 = g_3 + r_3p_3$

Et en généralisant :

➩ $r_i = g_{i-1} + r_{i-1}p_{i-1}$

b/ Exemple :

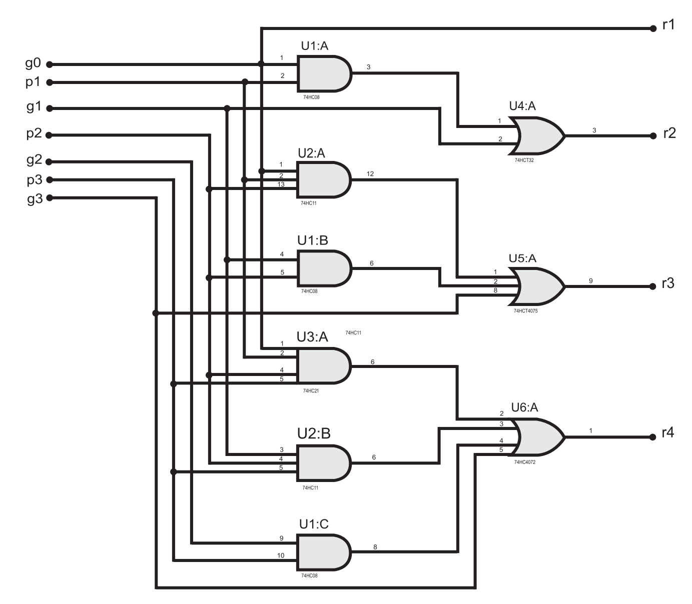
Etudions un cas particulier en supposant **i = 4** et bien entendu (r_0 **= 0**) ==> additionneur à **4 bits**.

Calculons les différentes retenues :

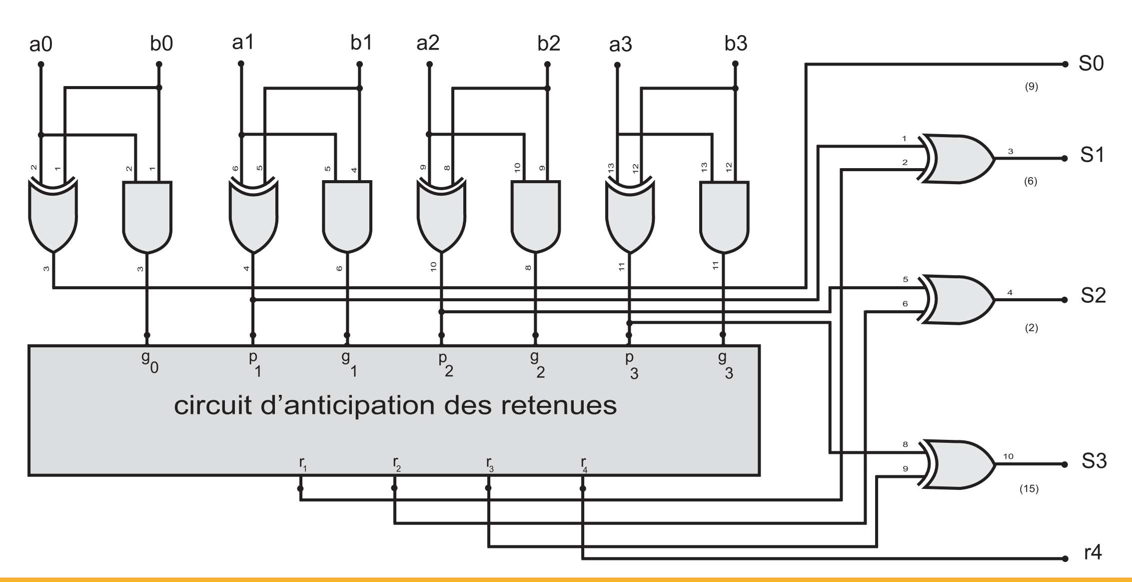
$r_1 = r_0p_0 + g_0 = g_0$
$r_2 = r_0p_0p_1 + g_0p_1 + g_1 = g_0p_1 + g_1$
$r_3 = r_0p_0p_1p_2 + g_0p_1p_2 + g_1p_2 + g_2 = g_0p_1p_2 + g_1p_2 + g_2$
$r_4 = r_0p_0p_1p_2p_3 + g_0p_1p_2p_3 + g_1p_2p_3 + g_2p_3 + g_3 = g_0p_1p_2p_3 + g_1p_2p_3 + g_2p_3 + g_3$

Avec $\mathbf{p_i = a_i + b_i}$ et $\mathbf{g_i = a_ib_i}$

D'où le circuit d'anticipation de l'additionneur **4 bits** :



Et finalement on obtient le schéma complet de l'additionneur **4 bits** avec son circuit d'anticipation des retenues :

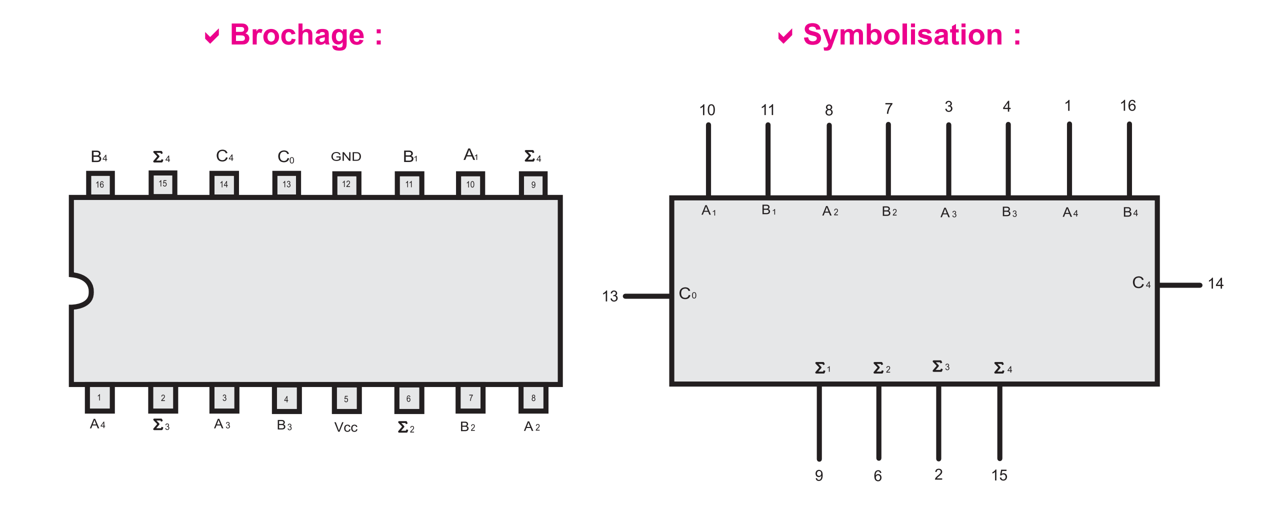




d/ Réalisation industrielle :

Parmi les circuits additionneurs intégrés, on cite celui de la **série 74XX83**

✔ **Brochage :** ✔ **Symbolisation :**



✔ **Structure interne d'un additionneur de deux nombres à 4 Bits :**

C0 A1 B1 A2 B2 A3 B3 A4 B4
13 10 11 8 7 3 4 1 16

VCC = PIN 5
GND = PIN 12

C1 C2 C3

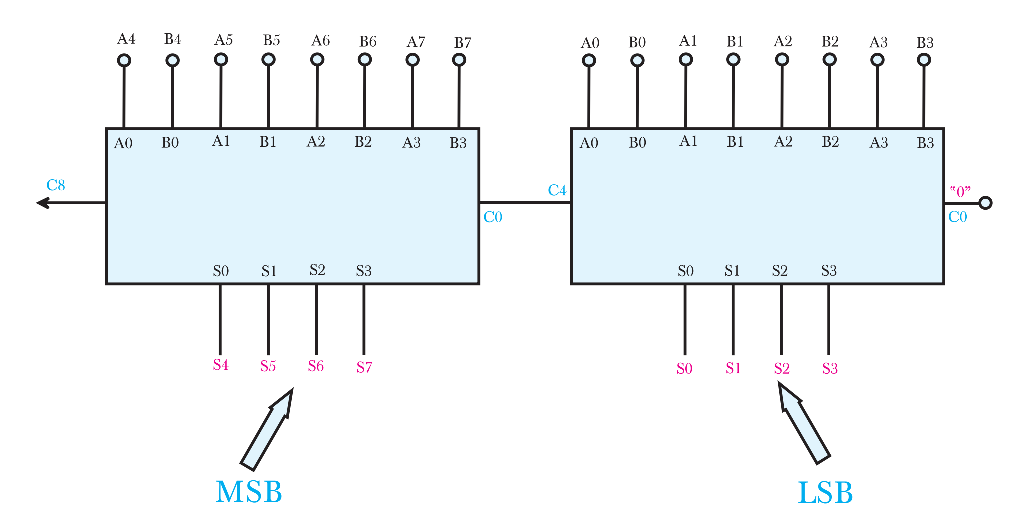
9 6 2 15 14
Σ1 Σ2 Σ3 Σ4 C4

Pour plus d'information : consultez le site http://www.datasheetcatalog.com

e/ Mise en cascade des additionneurs intégrés :

Pour additionner des nombres de plus de 4 bits, il faut utiliser plusieurs additionneurs de 4 Bits montés en cascade.

Exemple : addition de deux nombres à 8 bits ; cette application nécessite deux additionneurs montés de la façon suivante :



f/ L'additionneur et ses périphériques :

Dans les ordinateurs ou les systèmes de calcul, les nombres à manipuler, ainsi que les résultats des calculs, doivent être mémorisés ou stockés dans des endroits bien précis pour pouvoir les exploiter le moment opportun.

En plus de ces zones de stockages des périphériques supplémentaires (composants électroniques) sont nécessaires pour les préparatifs des calculs ou le transfert des données vers ou en provenance de l'élément de calcul qui peut être un additionneur, soustracteur, multiplieur, unité arithmétique et logique, microprocesseur, etc.

Activité

Réaliser l'activité **N°5** du **TP A1-1** dans le manuel d'activités : Le but de l'activité est la découverte des mécanismes mis en œuvre par un additionneur au cours de la réalisation d'une opération d'addition de deux nombres binaires, ainsi que l'exploration des périphériques nécessaires à cette opération.

1-6 Additionneur DCB "Décimal Codé Binaire" noté en anglais "BCD" :

a/ Principe :

Une notation fréquemment utilisée dans les ordinateurs est la notation **BCD**.

Dans cette notation, chaque chiffre du nombre décimal est représenté en binaire, soit sur **4 bits** (**BCD compacté**), soit sur **8 bits**.

Par exemple, en **BCD** compacté, $\mathbf{237_{10}}$ s'écrit donc : **0010 0011 0111**

À l'exception de certaines calculettes, peu d'ordinateurs utilisent cette notation, car elle prend plus d'espace mémoire que le binaire. En effet, toutes les combinaisons de bits supérieures à **1001** (**9**) sont inutilisées. Cependant, c'est une étape intermédiaire indispensable quand on veut passer du décimal au binaire ou du binaire au décimal.

Voici comment on peut effectuer l'addition en **BCD**. Si l'on additionne en arithmétique binaire les codes **BCD** correspondant à des chiffres dont la somme ne dépasse pas **9**, on aura évidemment un résultat donnant la représentation correcte de la somme en **BCD**. Par exemple, **64 + 21** donne bien **85**.
Cherchons en premier l'équivalent en **BCD** des deux nombres soit :

Décimal		BCD
64	⇒	0110 0100
21	⇒	0010 0001

D'où l'opération d'addition correspondante :

			Retenues	1	1						
+	6	4	+	0	1	1	0	0	1	0	0
	2	1		0	0	1	0	0	0	0	1
	8	5		1	0	0	0	0	1	0	1
					$8_{(10)}$				$5_{(10)}$		

Cependant, dès que la somme dépasse **9** dans un quartet de **4 bits**, il faut apporter une correction. Par exemple, si on additionne **7 + 5**, on obtient **12** en **décimal**, dont l'équivalent en **binaire** est : **1100**, or la valeur de ce quartet excède l'intervalle de définition d'un digit **BCD**. Donc pour remédier à ce problème on ajoute l'équivalent **binaire** de **6** au résultat trouvé et de cette façon on obtient bien la représentation exacte en **BCD** de la valeur recherchée (dans notre cas c'est le **12**).

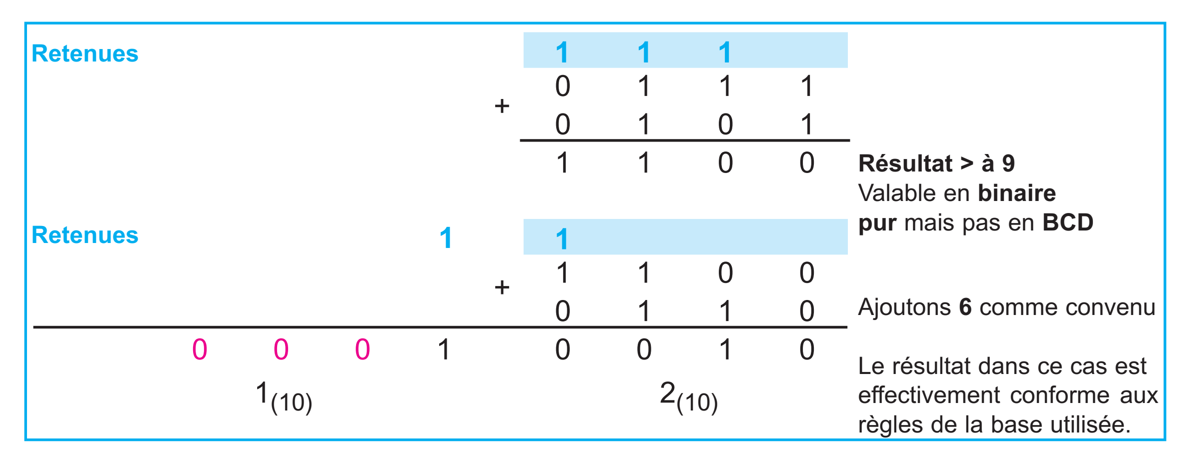
Sachant que :

Décimal		BCD
7	⇒	0000 0111
5	⇒	0000 0101
12	⇒	0001 0010

Or, si on effectue l'opération on obtient le résultat suivant qui n'est valable qu'en binaire pur.

	1		Retenues	1	1	1		
		7		0	1	1	1	
+		5	+	0	1	0	1	
	1	2		1	1	0	0	
					$12_{(10)}$			**Résultat > à 9** Valable en **binaire pur** mais pas en **BCD**

Donc pour remédier à ce problème on procède comme suit :
Chaque fois que le résultat d'un quartet est **supérieur** à **9** on ajoute $\mathbf{6_{(10)}}$ c'est-à-dire $\mathbf{0110_{(2)}}$ au résultat pour retrouver la transcription en **BCD** de ce résultat.
Reprenons notre exemple et ajoutons **6** comme convenu.

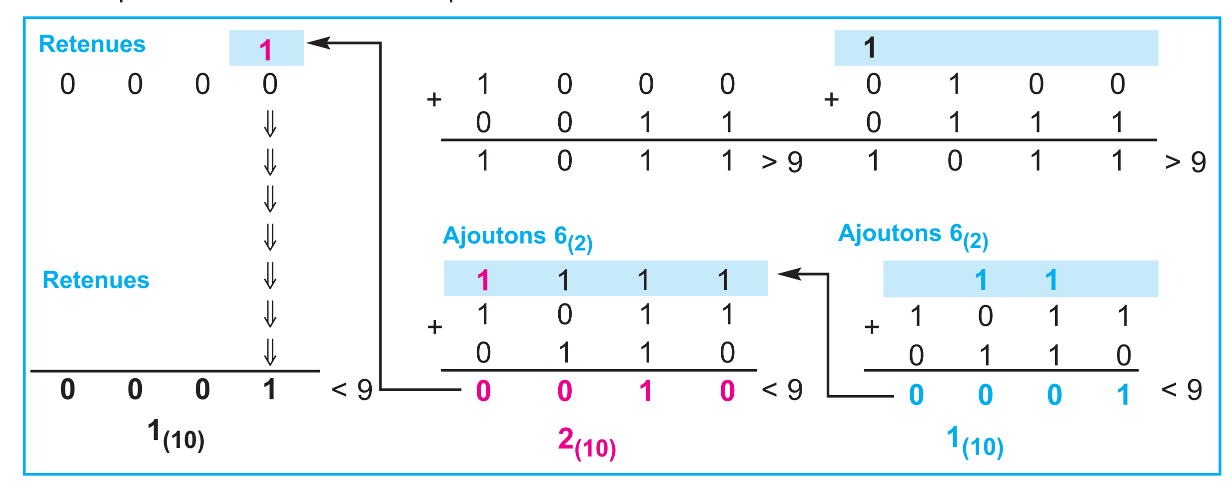


En conclusion : On effectue donc l'addition en quartets de **4 bits.** Si le résultat dépasse **9** pour l'un ou plusieurs de ces quartets, on leur ajoute **6** pour forcer une retenue et on obtient le résultat escompté en **BCD** :

Exemple : soit à additionner **$84_{(10)}$ + $37_{(10)}$ = $121_{(10)}$**

Décimal		**BCD**
8 4	⇒	1000 0100
+ 3 7	⇒	+ 0011 0111
1 2 1	⇒	0001 0010 0001

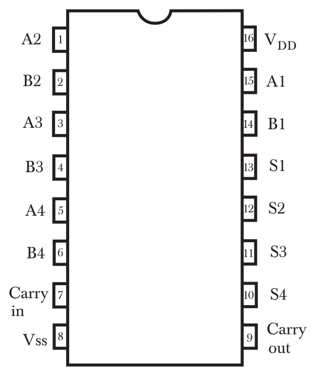
D'où l'opération d'addition correspondante :



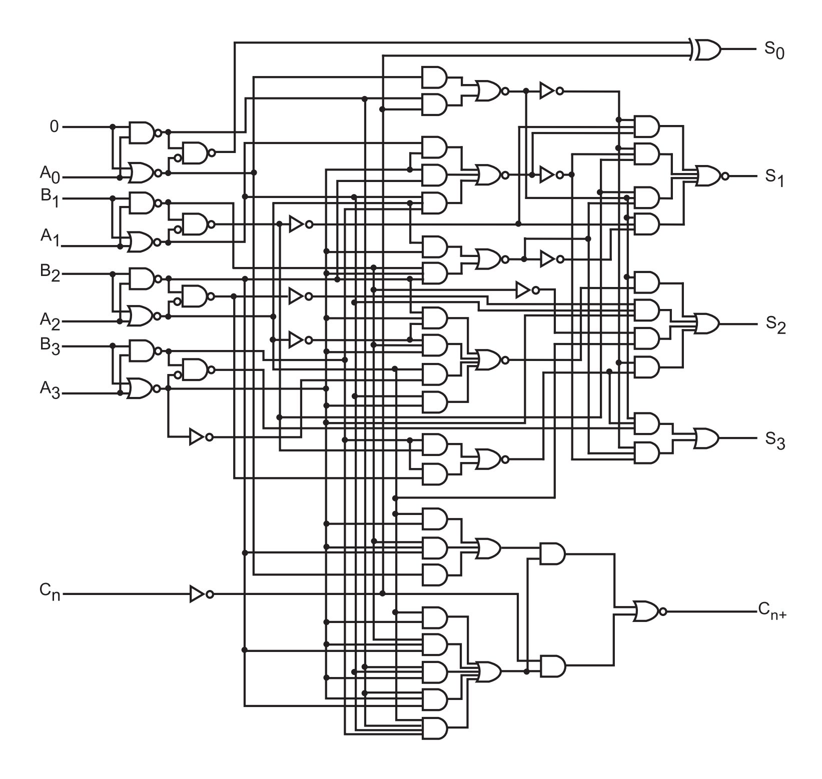
b/ Réalisation industrielle :

Exemple de circuit de la série **45XXX**

- **Brochage :**



Structure interne d'un additionneur BCD 4 bits :



Pour plus d'information : consulter le site http://www.datasheetcatalog.com

Activité
Réaliser l'activité **N°6** du **TP A1-1** dans le manuel d'activités : il s'agit de vérifier le fonctionnement d'un additionneur intégré de la famille **45xx**.

2- La multiplication ou le produit binaire :

Reprenons le système de gestion de parking, évoqué au début de cette leçon et essayons d'analyser la fonction **FP2.**
FP2 : permettre au propriétaire la gestion des sommes perçues.
Cette gestion peut nécessiter plusieurs opérations parmi les quelles la connaissance du total des sommes payées par les usagers durant une journée, une semaine, un mois, etc…
La connaissance de cette valeur nécessite une opération de multiplication du nombre des voitures garées durant une période bien définie par la somme due pour le stationnement d'une voiture, d'où la problématique :

Problème technique : Quelle est la structure électronique de base qui permet d'effectuer une opération de Multiplication ou Produit de deux nombres binaires ?

2-1 Principe :

Comme pour le système à base 10, la multiplication par 0 entraîne un résultat égal à 0.
La multiplication par 1 entraîne la recopie du multiplicande.
La procédure d'obtention du résultat est identique à celle de la multiplication décimale, le résultat est régi par les mêmes règles et principes qu'en décimal.
Sachant que la table de la multiplication est la suivante :

a_0	b_0	R	S
0	0	0	0
0	1	0	0
1	0	0	0
1	1	0	1

Découvrons les procédures de calcul sur un exemple.

Exemple : Soit à multiplier $25_{(10)}$ par $6_{(10)}$
On commence par transformer ces deux nombres en leur équivalent binaire

Décimal		Binaire
$25_{(10)}$	⇒	11001
$6_{(10)}$	⇒	110

Effectuons l'opération :

Retenues ⇓							
		X	1	1	0	0	1
					1	1	0
			0	0	0	0	0
1	1	1	1	0	0	1	-
	1	1	0	0	1	-	-
1	**0**	**0**	**1**	**0**	**1**	**1**	**0**
128	64	32	16	8	4	2	1

Pondération

Le résultat trouvé est évidemment : $150_{(10)}$ ==> $10010110_{(2)}$

2-2 Réalisation industrielle :

Exemple : multiplicateur binaire de deux nombres à 2 bits
Soit à réaliser la multiplication de deux nombres binaires à deux bits **A** (**A** : a_1a_0) et **B** (**B** : b_1 b_0).

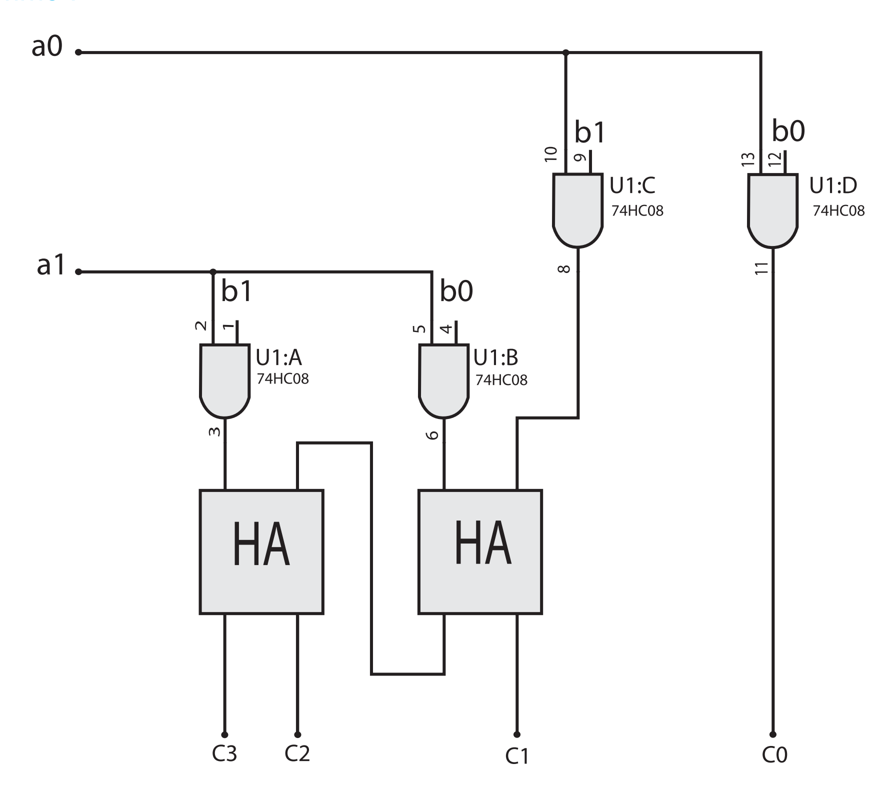
		a_1	a_0
		b_1	b_0
r_2	r_1	$a_1\ b_0$	$a_0\ b_0$
	$a_1\ b_1$	$a_0\ b_1$	
C_3	C_2	C_1	C_0

Réalisation du logigramme correspondant :

a/ Composants nécessaires :

Résultat	Composants nécessaires
C_0	Une fonction ET "AND"
C_1	Deux fonctions ET "AND" +1/2 Additionneur
C_2	Une fonction ET "AND" +1/2 Additionneur
C_3	

b/ Logigramme :



Activité

Réaliser l'activité **N°7** du **TP A1-1** dans le manuel d'activités : il s'agit de faire la synthèse d'un circuit multiplieur de deux nombres binaires à **3 bits**
(A :$a_2a_1a_0$) et **(B :$b_2b_1b_0$).**

2-3 La multiplication en complément à 2 :

a/ Si les opérandes sont positifs : la méthode est analogue à la multiplication décimale (résultat positif, bit de signe " 0 ").

b/ Si les deux opérandes sont négatifs : ils seront donc exprimés dans la notation en complément à 2. chacun de ces nombres est complémenté à 2 pour obtenir l'équivalent positif et ce sont les résultats de ces complémentations qu'on multiplie. Le produit résultant est un nombre positif dont le bit de signe est " 0 ".

c/ Si l'un des opérandes est positif et que l'autre est négatif : le nombre négatif est d'abord complémenté à 2 pour obtenir une grandeur positive.
Le produit est exprimé selon la notation en grandeur exacte. Cependant, ce produit doit être négatif car les nombres à multiplier sont de signes contraires. Par conséquent, on complémente à 2 le produit résultant et on ajoute le bit de signe " 1 " pour indiquer qu'il est négatif.

2-4 Le multiplieur en circuit intégré :

Parmi ces circuits on cite le **74284** et le **74285** de technologie **TTL**, ce circuit permet la multiplication de deux nombres à 4 bits.

Activité
Réaliser l'activité **N°8** du **TP A1-1** dans le manuel d'activités : il s'agit de simuler le fonctionnement un circuit multiplieur de deux nombres binaires à **4 bits** de la série **74xx**.



C- RESUMÉ

L'addition est l'opération arithmétique la plus importante dans les systèmes numériques. Les opérations de soustraction, de multiplication et de division effectuées par les systèmes numériques tels que l'ordinateur ne sont essentiellement que des variantes de l'opération d'addition.
Les machines numériques modernes, peuvent réaliser une opération d'addition en moins d'une microseconde.

Il est très important de saisir la différence entre l'addition logique de l'opération (**OU/OR**) et l'addition binaire.
L'addition logique est une opération booléenne effectuée par une porte (**OU/OR**), l'addition binaire est une opération arithmétique qui donne la somme arithmétique de deux nombres binaires.

Logique	Binaire (arithmétique)
1 + 1 = 1	1 + 1 = 0 avec report de **1**
1 + 1 + 1 = 1	1 + 1 + 1 = 1 avec report de **1**

Dans la notation en **DCB** ou **BCD**, la correction du résultat par l'ajout de l'équivalent binaire de $6_{(10)}$, n'aura lieu que dans les cas suivants :
- Si l'équivalent décimal du quartet dépasse **9.**
- Si la somme produit un report sur le quartet immédiatement à gauche.

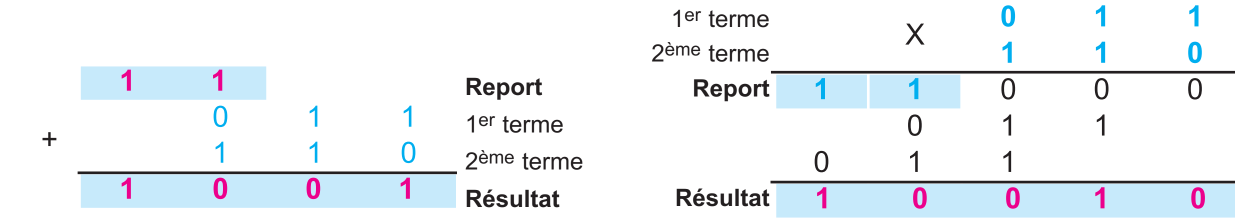
NB : pour mieux saisir le mécanisme, essayer d'additionner en BCD par exemple :
- **47 à 35.**
- **59 à 38.**

D- EXERCICES

1- Exercice résolu :

Exercice 1 :
Pour les paires des nombres binaires suivants (011,110), (1001,1111) et (10110,00111), effectuer les deux opérations d'addition et de multiplication.

Solution pour la première paire (011,110)



	1	1			Report
		0	1	1	1er terme
+		1	1	0	2ème terme
	1	0	0	1	Résultat

1er terme			0	1	1
2ème terme		X	1	1	0
Report	1	1	0	0	0
		0	1	1	
	0	1	1		
Résultat	1	0	0	1	0

2- Exercices à résoudre :

Exercice 1 :

Additionner les nombres décimaux suivants aprés les avoir convertis en binaire

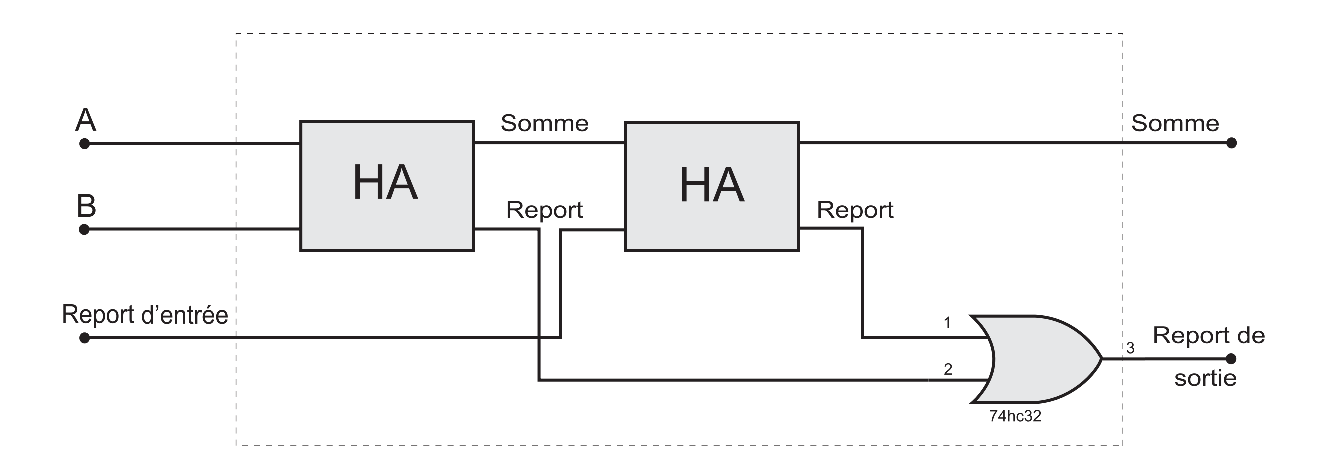
- a/ 64 + 13
- b/ 52 + 33
- c/ 145 + 382
- d/ 385 + 118

Multiplier les nombres décimaux suivants aprés les avoir convertis en binaire

- a/ 64 x 13
- b/ 52 x 33
- c/ 145 x 382

Exercice 2 :

Soit le logigramme suivant :



Sachant que **HA** est un demi-additionneur.

- a/ Simuler le fonctionnement de ce montage.
- b/ Etablir la table de vérité correspondante à ce logigramme.
- c/ Quelle fonction réalise ce logigramme.

Exercice 3 :

a/ Redémontrer que la somme **S** et la retenue **r** d'un additionneur à 2 bits prenant en entrée deux bits **A** et **B** et le report de l'additionneur précédent **R**, sont :

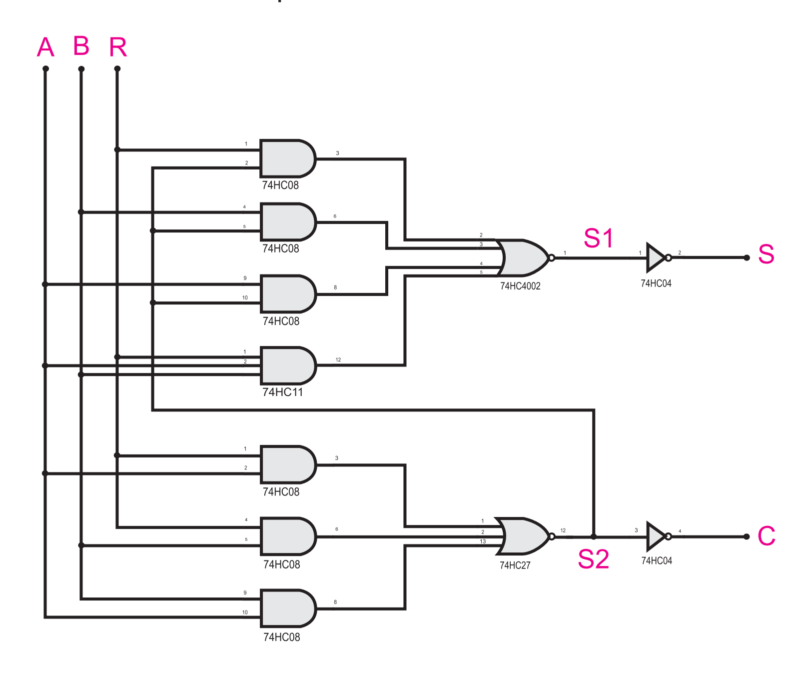
$$S = \overline{A}.\ \overline{B}.\ R + \overline{A}.\ B.\ \overline{R} + A.\ \overline{B}.\ \overline{R} + A.\ B.\ R$$

$$= A.\ B + A.\ R + B.\ R$$

b/ Montrer que l'on peut l'écrire sous cette forme :

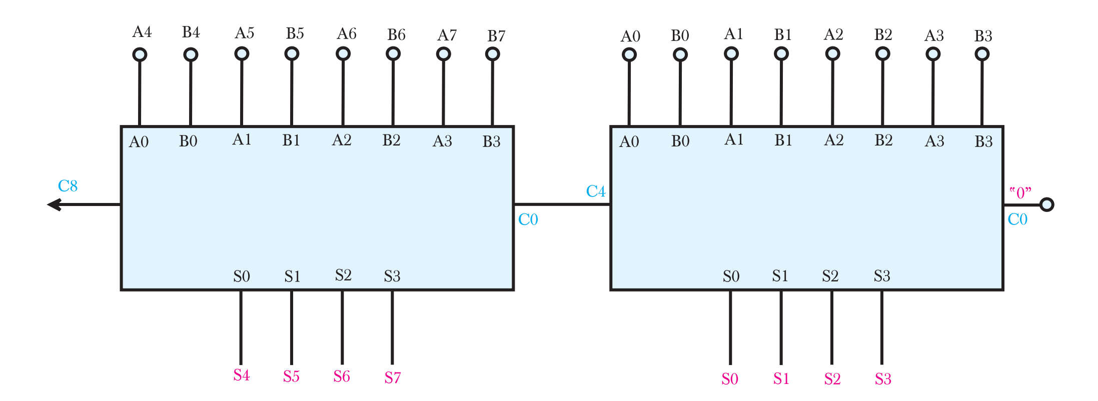
$$S = A \oplus B \oplus R$$

c/ Vérifier que le circuit suivant représente bien un tel additionneur.



Exercice 4 :

Déterminer les niveaux logiques aux entrées et aux sorties de l'additionneur **8 bits** de la figure suivante, quand $137_{(10)}$ est additionnée à $72_{(10)}$

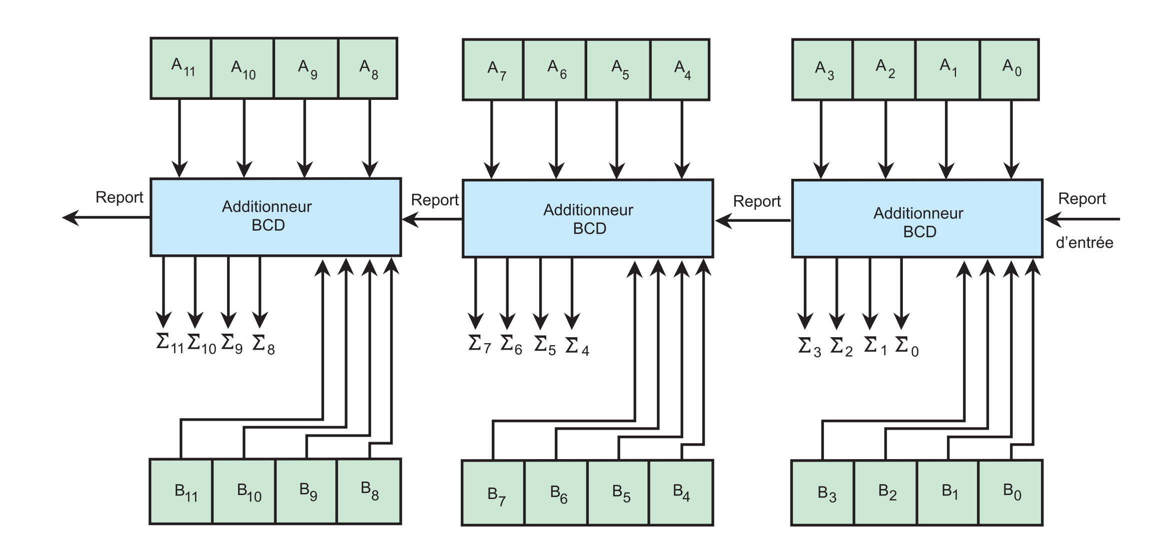


Exercice 5 :

Représenter les valeurs décimales suivantes : 135 et 265 en BCD, additionner ces deux valeurs. Vérifier le résultat trouvé en le reconvertissant en décimal.

Exercice 6 :

Soit le schéma suivant :

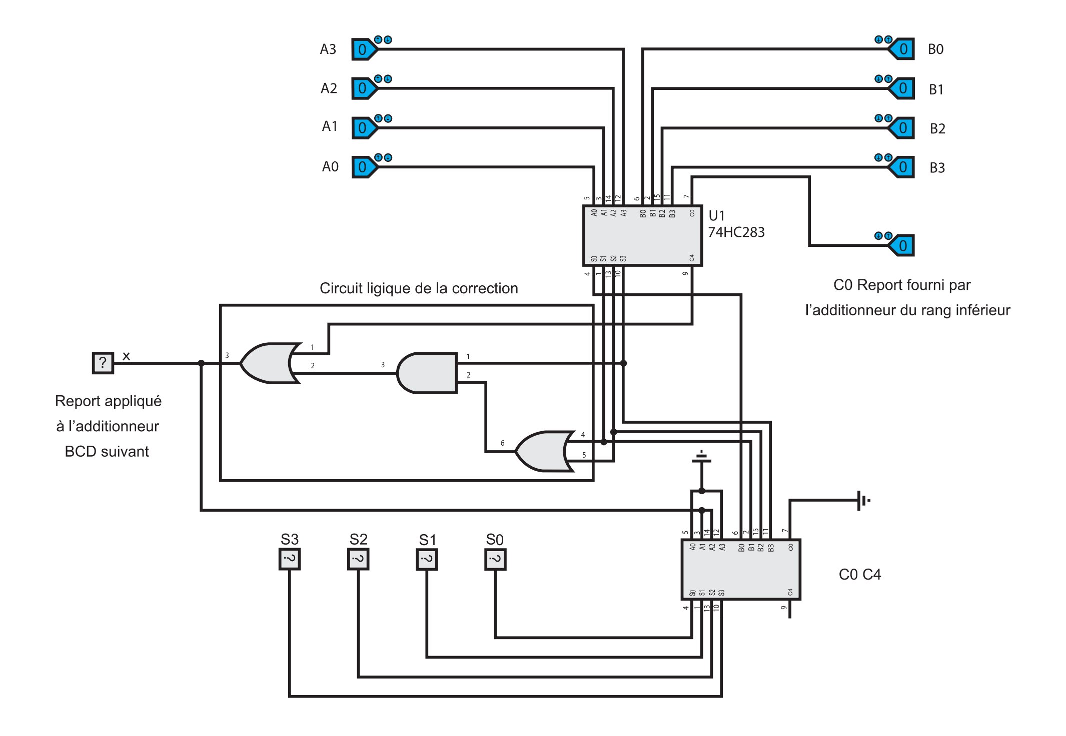


– Proposer un nom pour ce montage ?

– Déterminer les niveaux logiques des entrées et des sorties de ce circuit lors de l'addition des deux chiffres décimaux suivants : **$247_{(10)}$** et **$538_{(10)}$**.

Exercice 7 :

Le logigramme suivant, représente un additionneur DCB.



Lors de sa mise en œuvre, on relève les niveaux suivants :

Cas	Opérande B				Opérande A				Résultat				Report (X)
	B3	B2	B1	B0	A3	A2	A1	A0	S3	S2	S1	S0	
1	0	0	1	1	0	1	1	0	1	0	0	1	0
2	0	1	1	1	1	0	0	0	1	1	1	1	0
3	1	0	0	1	1	0	0	1	0	0	1	0	0

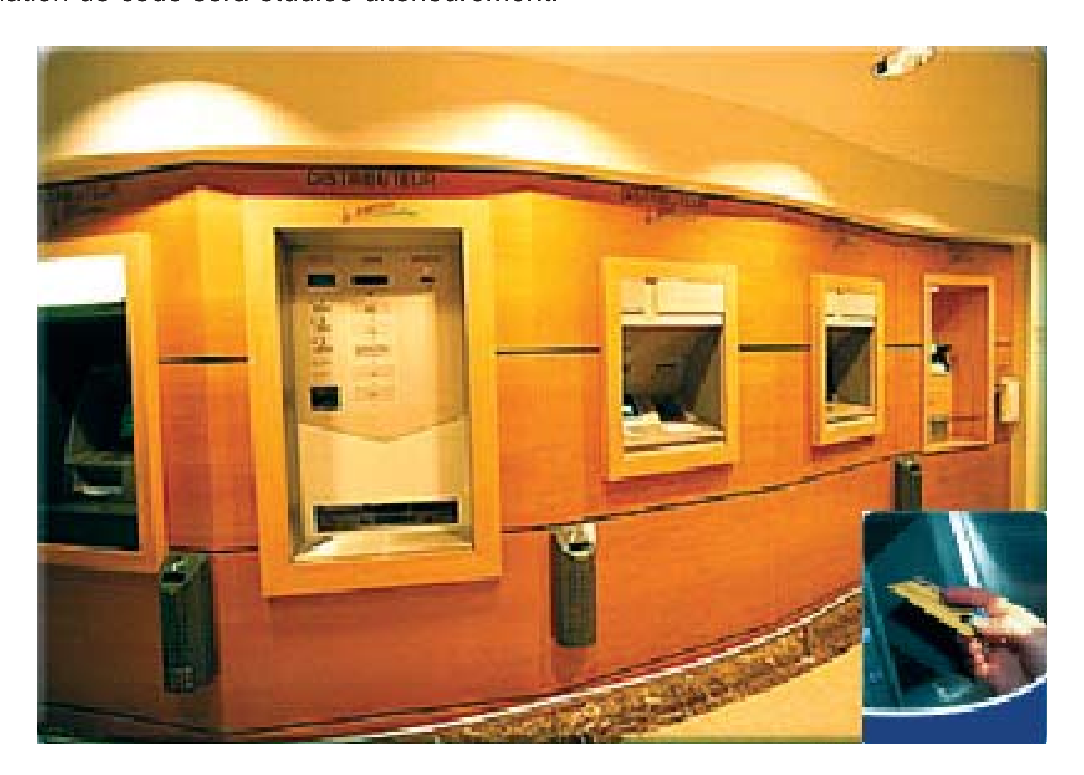
a/ Vérifier si les résultats trouvés sont justes.
b/ En cas d'erreur, donner le résultat que doit fournir l'additionneur.
c/ Expliquer le cas échéant la ou les causes de la défaillance.

NB : pistes à exploiter, il se peut que :

- La sortie de la porte (**AND / ET**) est maintenue au niveau **"0"**
- Les entrées de l'une des portes (**OR / OU**) sont court-circuitées.
- La liaison entre l'additionneur U2 et le point "X" est coupée.

E- PROBLEME DE SYNTHESE

Un usager souhaite retirer de l'argent ; pour cela il introduit sa carte bancaire dans un distributeur automatique de billets ; on distinguera une étape de reconnaissance du code personnel, puis une étape de distribution de la somme demandée par l'usager.
Les temporisations nécessaires au bon fonctionnement du système ne seront pas prises en compte.
L'étude se limitera à l'étape de distribution seulement ; l'étape de reconnaissance et de validation de code sera étudiée ultérieurement.



Etape de distribution :

Le système est muni d'un clavier à **3 touches**, notées " **a** ", " **b** " et " **c** ". Ces touches permettent l'activation respective des liasses de **10 D**, **20 D** et **30 D**.
Le système permet de délivrer les sommes suivantes : **10 D**, **20 D**, **30 D**, **40 D**, **50 D** et **60 D** en ne prenant en compte qu'une seule frappe par touche et en faisant l'addition éventuellement.

Travail demandé :

a/ Dresser la table de vérité des sommes choisies en fonction des entrées " **a** ", " **b** " et " **c** ".
b/ Déduire les différentes équations logiques.
c/ Etablir le logigramme correspondant avec un ordre de validation " **v** ".

LA SOUSTRACTION ET LA DIVISION EN BINAIRE

A- LA SOUSTRACTION BINAIRE

Reprenons le système “gestion d’un parking”. On désire trouver une solution pour la fonction FC1, dont l’intitulé est le suivant :
FC1 : Signaler à l’usager que les places disponibles sont épuisées.
La réponse à cette contrainte, nécessite la connaissance de la différence entre le nombre des véhicules entrants et le nombre des véhicules sortants, d’où une structure électronique bien adaptée pouvant répondre à la problématique posée :

Problème technique :

Quelle est la structure électronique de base qui permet d'effectuer une opération de soustraction entre deux nombres binaires ?

1- Principe :

Le principe reste le même qu’en décimal. On retranche, dans la colonne de poids le plus faible, le chiffre soustracteur du chiffre soustrait, autrement dit on prend le **complément** du chiffre soustracteur par rapport au chiffre soustrait.
Si le chiffre soustrait a une valeur numérique plus faible que celle du chiffre soustracteur, il y a emprunt au terme soustrait de la colonne de poids immédiatement supérieur.
On procède ainsi de colonne en colonne jusqu’à la dernière représentant le poids le plus élevé.
La table de la soustraction binaire pour deux bits est régie par des règles strictes comme suit :

a_0	b_0	r	D
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

Exemple : soit à soustraire **43** de **67**
$67_{(10)}$ ==> $1000011_{(2)}$
$43_{(10)}$ ==> $101011_{(2)}$
Ainsi, on obtient : $11000_{(2)}$ ==> $24_{(10)}$

Pondération

	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
	1	**0**	**0**	**0**	**0**	**1**	**1**	1er terme
–		**1**	**0**	**1**	**0**	**1**	**1**	2ème terme
Reports	**1**	**1**	**1**					
	0	**0**	**1**	**1**	**0**	**0**	**0**	**Résultat**

2- Réalisation pratique :

2-1 Le demi-soustracteur :

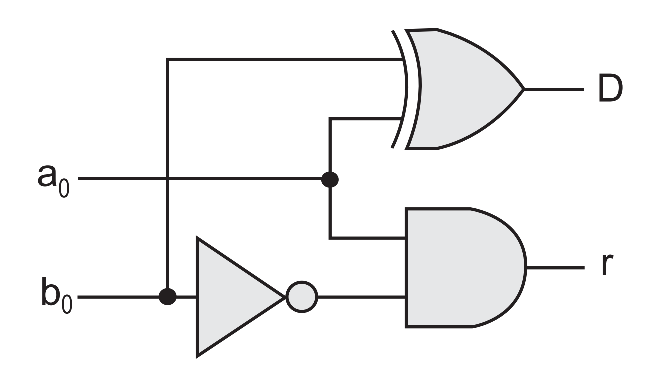
Pour une soustraction entre deux nombres **A (A = a_0)** et **B (B = b_0)** de **1** bit, **4** combinaisons sont possibles, et le résultat occupera **2 bits**, un bit pour la différence **(D)** et un autre pour la retenue **(r).** Ce dispositif est également appelé demi-soustracteur.
Sa table de vérité et ses expressions logiques sont les suivantes :

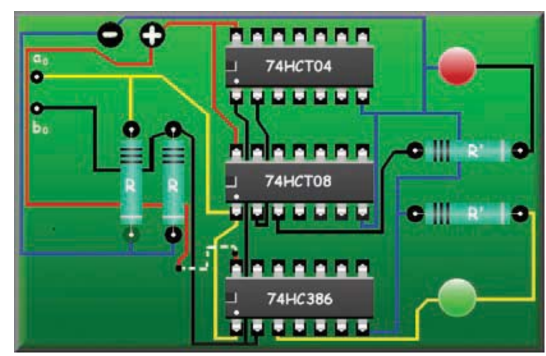
a_0	b_0	r	D
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

$$D = a_0 \oplus b_0$$

$$r = \overline{a_0} . b_0$$

Le logigramme correspondant est comme suit :





Activité
Réaliser l'activité N°1 du TP-A1-2 dans le manuel d'activités.
Il s'agit de cabler ou simuler le fonctionnement d'un circuit à base de cellules logiques, de tracer sa table de fonctionnement et de chercher les équations des sorties en vue de vérifier qu'il s'agit d'un demi-soustracteur.

2-2 Le soustracteur complet de deux nombres à 1 bit :

Nous pouvons généraliser cette structure pour décrire la soustraction de mots de taille supérieure à 1. Pour cela il faut introduire une variable supplémentaire r_i qui représente une retenue entrante.

La table de vérité, résumant le fonctionnement du soustracteur « 1 bit » avec retenue entrante (ou soustracteur « complet ») est comme suit :

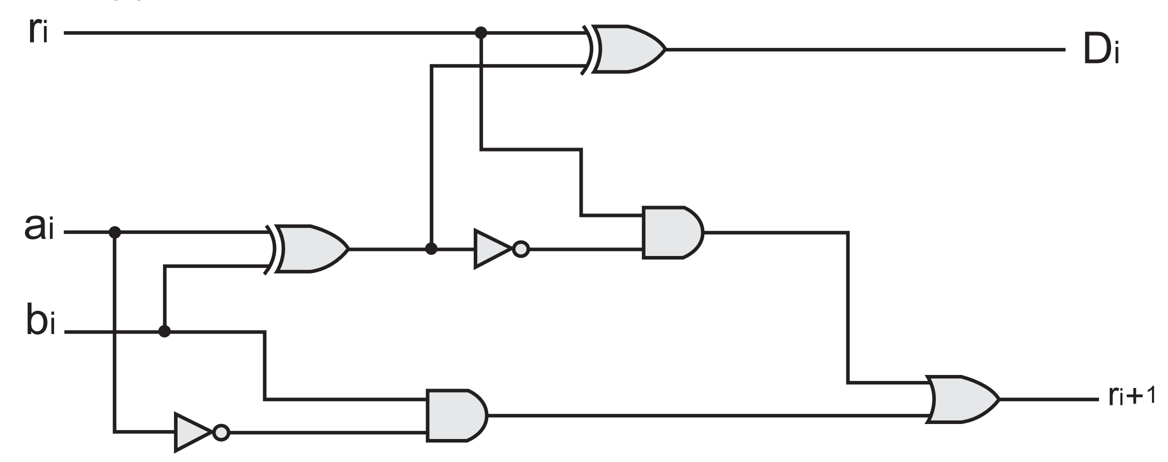
a_i	b_i	r_i	r_{i+1}	D_i	Base 10
0	0	0	0	0	0
0	0	1	1	1	-1
0	1	0	1	1	-1
0	1	1	0	1	-2
1	0	0	1	0	1
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	1	1	-1

Un raisonnement identique à celui utilisé dans le cas de l'additionneur aboutit aux équations :

$$D_i = a_i \oplus b_i \oplus r_i$$

$$r_{i+1} = \overline{a_i}.\, b_i + r_i.\, \overline{(a_i \oplus b_i)}$$

D'où le logigramme correspondant :



Activité

Réaliser l'activité N°2 du TP-A1-2 dans le manuel d'activités

Il s'agit de cabler ou simuler le fonctionnement d'un circuit à base de cellules logiques, de tracer sa table de fonctionnement et de chercher les équations des sorties en vue de vérifier qu'il s'agit d'un soustracteur complet.

NB : Cette méthode demeure valable mais difficile à mettre en œuvre. Pour contourner ces difficultés, on peut exploiter les propriétés du complément en vue de ramener l'opération soustraction à une simple opération d'addition.

3- Le complément d'un nombre :

Exemple : le complément de 35 par rapport à 53 est égale à : 53 – 35 = 18
Or cette notion de complément n'est intéressante que dans des cas bien particuliers dont la finalité est généralement la réalisation d'opérations dans les systèmes numériques (dans ce cas on cite, le complément à 9, le complément à 10 en décimal et le complément à 1 et le complément à 2 dans le système binaire).

3-1 Le complément à 1 : le complément à 1 d'un nombre binaire est la valeur numérique qu'il faut ajouter à ce nombre pour obtenir la valeur numérique immédiatement inférieure à celle de la puissance supérieure.

Exemple : Soit à chercher le complément à **1** de **10011**.
La puissance immédiatement supérieure à **10011** est **100000**.
La valeur numérique immédiatement inférieure est : **11111**
Posons l'opération (**11111**) – (**10011**) soit :

	1	**1**	**1**	**1**	**1**
–	**1**	**0**	**0**	**1**	**1**
	0	**1**	**1**	**0**	**0**

Le nombre binaire **01100** est le complément à **1** de **10011** ; si on additionne ces deux nombres nous obtenons le nombre binaire **11111**.

Une solution pratique, consiste à remplacer les **0** par des **1** et vice-versa pour trouver le complément à **1** d'un nombre binaire quelconque.

3-2 Le complément à 2 :

Le complément à 2 d'un nombre binaire est la valeur qu'il faut ajouter à un nombre fixé d'avance pour obtenir la valeur de la puissance immédiatement supérieure.
Une solution pratique consiste à chercher en premier le complément à **1** et à ajouter à ce dernier un **1** binaire.
Exemple : soit à déterminer le complément à **2** du nombre binaire **10011**.
complément à **1** correspondant **01100**
complément à **2** : (**01100**) + (**1**)

	1	**0**	**0**	**1**	**1**	Nombre de départ
	0	**1**	**1**	**0**	0	Complément à 1
+					**1**	Ajout d'un 1
	0	**1**	**1**	**0**	**1**	**Complément à 2**

Le nombre binaire **01101** est le complément à **2** de **10011**
Le même résultat peut être obtenu en retranchant le nombre binaire de la puissance immédiatement supérieure.

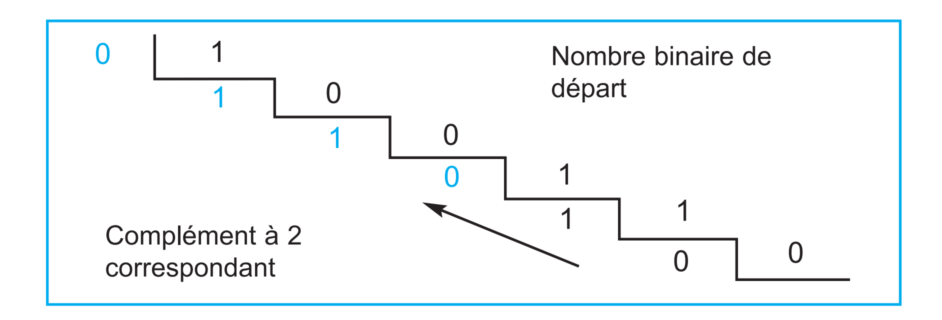
Une solution pratique consiste à retranscrire le nombre dont on cherche le complément à **2** en partant de la droite (poids le plus faible) sans aucun changement jusqu'au premier **1** rencontré, puis à inverser systématiquement les uns et les zéros rencontrés.

Exemple : soit à chercher le complément à **2** du nombre binaire **100110**

✔ 1ère solution en passant par le complément à **1**

	1	0	**0**	**1**	**1**	**0**	Nombre de départ
	0	1	**1**	**0**	**0**	1	Complément à 1
+						1	Ajout d'un 1
Reports					**1**		
	0	**1**	**1**	**0**	**1**	**0**	**Complément à 2**

✔ 2ème solution sans passer par le complément à **1**



NB : toute cette gymnastique mathématique binaire, va servir à réaliser les différentes opérations dans les systèmes numériques en particulier la soustraction qui va être ramenée à une simple addition comme suit :

X = a – b

X = a + (complément à 2 de b)

Pour le premier exemple : soit à soustraire **43** de **67**

Décimal		**Binaire**
6 7	⇒	1000011
– 4 3	⇒	– 101011
2 4	⇒	11000



On peut vérifier le résultat suivant :

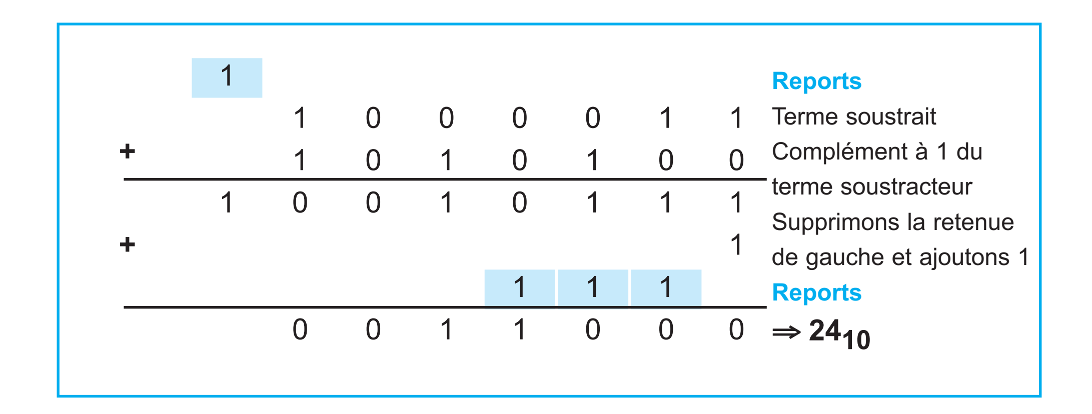
67 $_{(10)}$ **- 43** $_{(10)}$ **= 24** $_{(10)}$

1000011 $_{(2)}$ **- 101011** $_{(2)}$ **) = 0011000** $_{(2)}$

Pour chercher le complément à **2** de **101011** $_{(2)}$

Soit : **010101** $_{(2)}$

On fait maintenant la somme : **1000011** $_{(2)}$ **+ 010101** $_{(2)}$ **= 11000**



4- La soustraction en complément à 2 :

Soit à effectuer **A-B**, la marche à suivre pour réaliser cette opération est comme suit :

a/ prendre le **complément à 2** de **B**, y compris le bit de signe

- Si **B** est positif, il devient négatif.
- Si **B** est négatif, sa complémentation à 2, en fera un nombre positif écrit en grandeur exacte. Autrement dit, nous changeons le signe de **B.**

b/ Après complémentation à 2, on additionne l'ensemble (**A** conserve sa forme initiale). Le résultat représente la différence recherchée. Le bit de signe de la différence, informe si la réponse est positive ou négative et si on est en notation binaire exacte ou en notation en complément à 2.

NB : Les deux nombres doivent avoir le même nombre de bits.

Activité

Réaliser l’activité N°3 du TP-A1-2 dans le manuel d’activités

Il s'agit de cabler ou simuler le fonctionnement d’un circuit Additionneur/Soustracteur parallèle dans la notation en complément à 2 et à découvrir les périphériques de ce circuit.

B- LA DIVISION OU LE QUOTIENT BINAIRE

Le cheminement en vue de l'obtention du résultat est identique à celui de la division décimale. Par conséquent et pour illustrer cette opération nous allons passer à un exemple.
La procédure pratique implique deux opérations :

la multiplication ou produit
la soustraction ou différence.

Ces opérations nous sont familières, leurs tables sont représentées plus haut dans ce cours.

Exemple :
Soit à diviser **10100101_2 (165_{10})** par **1011_2 (15_{10}).**

Division décimale	Division binaire
1 6 5 | 1 1 – 1 1 | 1 5 5 5 – 5 5 0	– 1 0 1 0 0 1 0 1 | 1 0 1 1 0 0 0 0 | 0 1 1 1 1 1 0 1 0 0 – 1 0 1 1 1 0 0 1 1 – 1 0 1 1 1 0 0 0 0 – 1 0 1 1 0 1 0 1 1 – 1 0 1 1 0 0 0 0

C- RESUMÉ

Ecriture d'un nombre signé dans la notation en complément à 2 :

- Si le nombre est positif :
  - ✓ Grandeur = grandeur binaire exacte.
  - ✓ Bit de signe = 0 à placer devant le bit de poids le plus fort.
- Si le nombre est négatif :
  - ✓ Grandeur = complément à 2 de la grandeur exacte.
  - ✓ Bit de signe = 1 à placer devant le bit de poids le plus fort.

Exemple :

✓ 45(10) = 101101(2)
✓ Le complément à 1 de 45(10) = 010010
+ 1
✓ Le complément à 2 de 45(10) = 010011

D'où :

0	1	0	1	1	0	1	$+ 45_{(10)}$
1	0	1	0	0	1	1	$- 45_{(10)}$

La notation en complément à **2** d'un nombre signé, transforme un nombre positif en un nombre négatif et vice versa.
Dans la notation en complément à **2**, les nombres à traiter doivent avoir le même nombre de bits.
Le bit de signe, doit être traité sur le même pied d'égalité que les bits de la partie grandeur.
La notation en complément à **2** est très utilisée dans les machines numériques du fait qu'avec le même circuit, on peut additionner et soustraire.

D- EXERCICES

1- Exercice résolu :

Exercice 1 :

Représenter chacun des nombres décimaux signés suivants, selon la notation en complément à 2. utiliser un total de 8 Bits y compris le bit de signe : «+32», «-14», «+63», «-104», «-1», «-128», «+169» et «0».

Solution pour les deux premiers nombres «+32» et «-14»

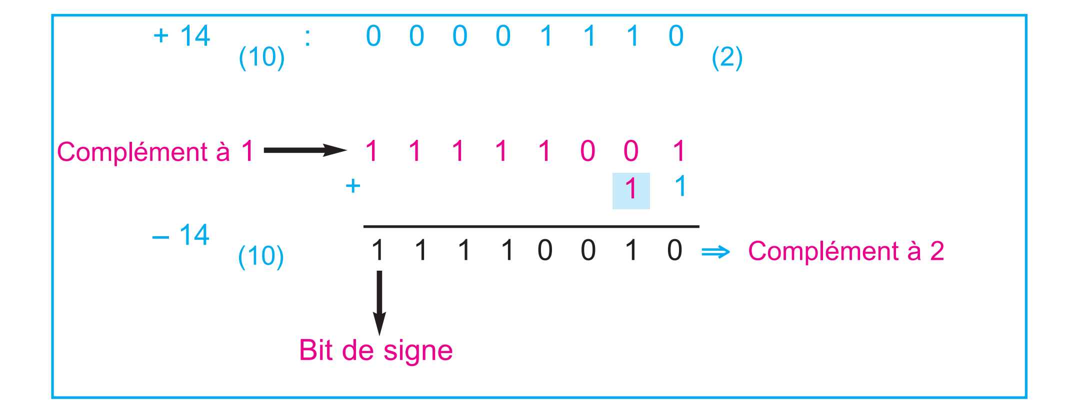
– «+32» est un nombre positif, la grandeur **32** est la grandeur exacte, soit sur 8 bits :

$+32_{(10)}$: 0 0 1 0 0 0 0 0 $_{(2)}$

⇓

Bit de signe

– «-14» est un nombre négatif, la grandeur **14** est écrite au moyen du complément à **2** comme suit :





2- Exercices à résoudre :

Exercice 1 :

Pour représenter les nombres signés, on peut adopter la notation «signe-grandeur» ou «signe-norme» qui se résume comme suit :

- signe : représenté par un seul bit «0 pour un nombre positif» et «1 pour un nombre négatif».
- norme ou grandeur : représentée par un ensemble de bits formant l'équivalent binaire du nombre décimal traité.

En se basant sur cette constatation, trouver l'équivalent décimal de chacune des représentations en complément à 2 des nombres binaires signés suivants : «01100», «11010» et «10001».

Exercice 2 :

Déduire la gamme ou l'intervalle des valeurs décimales signées qu'on peut représenter avec 12 Bits y compris le bit de signe.

Exercice 3 :

Donner le nombre de bits nécessaires pour représenter les nombres décimaux compris entre «-32768» et «+32767».

Exercice 4 :

Effectuer les opérations suivantes avec la notation en complément à 2. Utiliser pour chaque nombre 8 Bits (y compris le bit de signe) : soustraire : «+16 de +17», «+47 de +47», «+21 de -13» et «-36 de -15».

Exercice 5 :

Effectuer les divisions suivantes : «1100 / 100», «111111 / 1001», et «10111 / 100».

Exercice 6 :

Sachant que quand un nombre signé a « 1 » comme bit de signe et partout des «0» comme bits de grandeur, son équivalent décimal est «-2^n», où «N» est le nombre de bits de grandeur.

Déterminer l'équivalent décimal des nombres binaires signés suivants : «100», «1000» et «10000».

Exercice 7 :

Rappeler les principes d'un demi-additionneur puis d'un additionneur complet ; déduire de ces principes un circuit logique permettant d'implémenter le complément à 2 sur n bits.

Proposer un circuit matérialisant une décrémentation à n bits.

Exercice 8 :

Réaliser un soustracteur complet en respectant les contraintes suivantes :

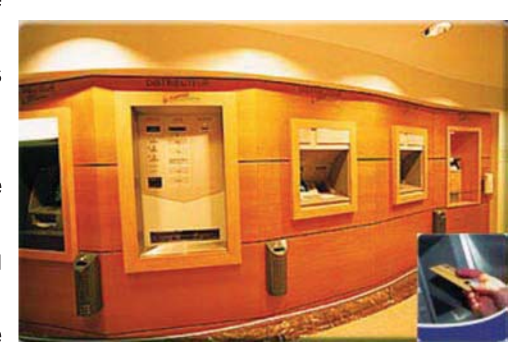
- avec deux demi-soustracteurs.
- avec un demi soustracteur et un demi additionneur.

LES COMPARATEURS LOGIQUES

A- MISE EN SITUATION

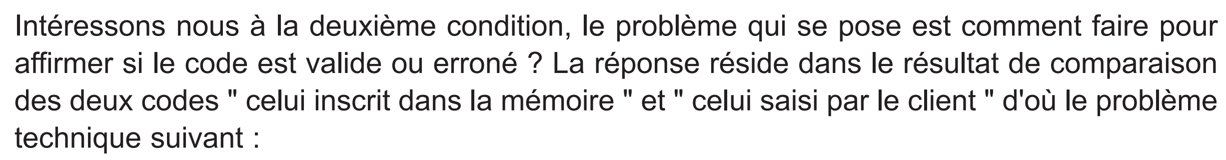
1- Fonctionnement du système : distributeur automatique de billets

Reprenons le système **distributeur automatique de billets.**



Etudions avec plus de détails l'étape de reconnaissance du code secret.
Le code mis en mémoire " B " est inscrit sur **2 bits** notés " b_1 " et " b_0 " en binaire pur.
Un clavier de saisie mis à la disposition de l'utilisateur, permet de taper son code secret " A " ; ce clavier de saisie envoie ce code sur **2 variables** " a_1 " et " a_0 ".
L'attribution de la somme requise n'aura lieu que si au moins les conditions suivantes sont remplies :

- Le solde client permet l'attribution de la somme demandée.
- Le code client est valide.



Intéressons nous à la deuxième condition, le problème qui se pose est comment faire pour affirmer si le code est valide ou erroné ? La réponse réside dans le résultat de comparaison des deux codes " celui inscrit dans la mémoire " et " celui saisi par le client " d'où le problème technique suivant :

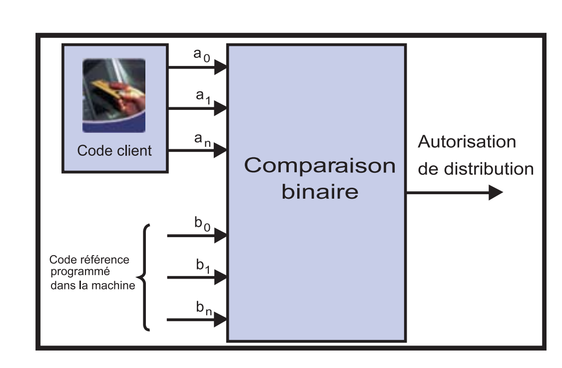


Problème technique : Quelle est la structure électronique de base qui permet d'effectuer une opération de comparaison de deux nombres binaires ?

B- SYNTHESE DES COMPARATEURS

1- Principe :

Le principe général de la comparaison de deux nombres codés en binaire :
A et B, indique en sortie si :
A = B, A > B ou si A < B.



A = $a_n ... a_3a_2a_1a_0$ et **B =** $b_n ... b_3b_2b_1b_0$
La première étape consiste à comparer les bits du poids le plus fort. Dans notre cas de figure a_n et b_n.
Trois cas peuvent se présenter :

- $a_n < b_n$ ⇨ $A < B$ quel que soit l'état des autres bits de A et de B.

Ou

- $a_n > b_n$ ⇨ $A > B$ quel que soit l'état des autres bits de A et de B.



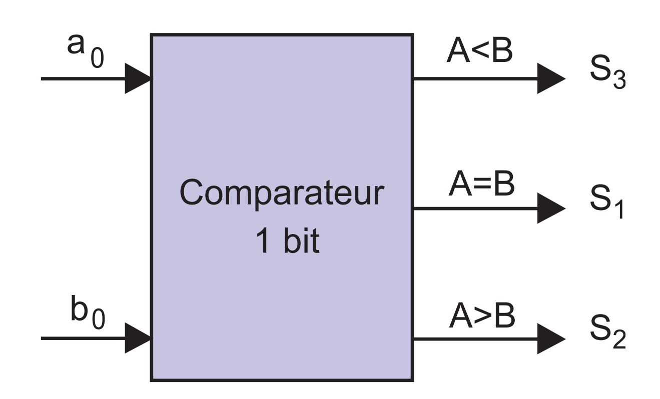
✔ a_n = b_n ⇨ dans ce cas nous ne pouvons conclure, et il faut poursuivre la comparaison sur les bits du poids inférieur a_{n-1} et b_{n-1}.
Dans certains cas, on ne peut conclure que lors de la comparaison des bits du poids le plus faible soient " a_0 et b_0 " ; d'où la nécessite d'un comparateur élémentaire à un bit.

2- Comparateur 1 bit :

2-1 Principe :

Ce comparateur permet la comparaison de deux nombres codés chacun sur un seul bit.

Soit : $A = a_0$ $B = b_0$



2-2 Table de vérité :

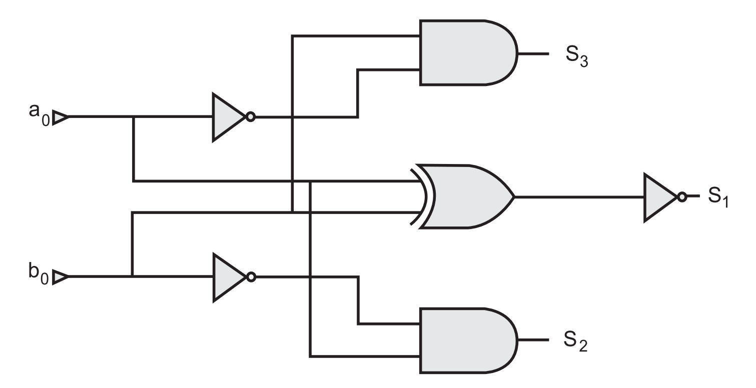
a_0	b_0	S_1	S_2	S_3
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

$S_1=a_0b_0+\overline{a_0b_0} = \overline{a_0 \oplus b_0}$ $S_2=a_0\overline{b_0}$ $S_3=\overline{a_0}b_0$

Légende :

S_1 : A = B
S_2 : A > B
S_3 : A < B

2-3 Logigramme :



Activité
Réaliser l'activité **N°1** du **TP A2-1** dans le manuel d'activités : cette activité consiste à comparer deux nombres binaires à un bit chacun (**A** : a_0) et (**B** : b_0) et de signaler à la sortie si A<B, A>B ou A=B au moyen des sorties respectives : S_3, S_2 et S_1.

3- Généralisation :

Pour comparer deux nombres codés sur plusieurs bits, il est nécessaire d'associer entre eux autant de comparateurs élémentaires qu'il y a de bits.
Il faut donc généraliser à n bits les conclusions et les équations établies précédemment.

Soit à comparer : $A = a_3.a_2.a_1.a_0$ et $B = b_3.b_2.b_1.b_0$

- Pour conclure que **A = B**, il faut que :
  $a_3 = b_3$ ET $a_2 = b_2$ ET $a_1 = b_1$ ET $a_0 = b_0$
  Donc :

$$S_1 = (a_0 \odot b_0).(a_1 \odot b_1).(a_2 \odot b_2).(a_3 \odot b_3)$$

- Pour conclure que **A > B**, il faut que :
  ($a_3 > b_3$) OU ($a_3 = b_3$ ET $a_2 > b_2$) OU ($a_3 = b_3$ ET $a_2 = b_2$ ET $a1 > b_1$)
  OU ($a_3 = b_3$ ET $a_2 = b_2$ ET $a_1 = b_1$ ET $a_0 > b_0$)
  Donc :

$$S_2 = a_3 \odot \overline{b_3} + (a_3 \odot b_3).a_2\overline{b_2} + (a_3 \odot b_3).(a_2 \odot b_2).a_1\overline{b_1} + (a_3 \odot b_3).(a_2 \odot b_2).(a_1 \odot b_1).a_0\overline{b_0}$$

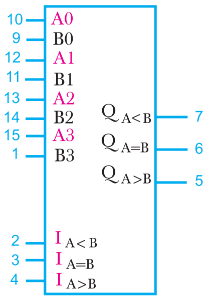
- Pour conclure que **A < B**, il suffit de remplacer dans la démarche ci-dessus le symbole **>** par **<**, d'où :

$$S_3 = \overline{a_3}b_3 + (a_3 \odot b_3).\overline{a_2}b_2 + (a_3 \odot b_3).(a_2 \odot b_2).\overline{a_1}b_1 + (a_3 \odot b_3).(a_2 \odot b_2).(a_1 \odot b_1).\overline{a_0}b_0$$

4- Comparateurs intégrés :

4-1 Exemple de circuit :

Comparateur de deux nombres à **4 Bits** de la famille **XX85**.



Symbole 01

Symbole 02

Ce circuit standard, compare deux mots " **A** " et " **B** ", de **4 bits** chacun. Outre les entrées de données recevant les deux mots à comparer " **A** : broches **10**, **12**, **13** et **15** " et " **B** : broches **9**, **11**, **14** et **1** ", il possède également trois autres entrées " **A > B** broche **4** ", " **A = B** broche **3** " et " **A < B** broche **2** ", permettant la mise en cascade des comparateurs pour pouvoir comparer des nombres de plus de **4 Bits**.
Si le circuit est utilisé seul " comparaison de deux nombres de **4 bits** au plus ", les entrées " **A > B** ", " **A = B** " et " **A < B** " doivent être connectées respectivement aux niveaux logiques " **0** ", " **1** " et " **0** ".

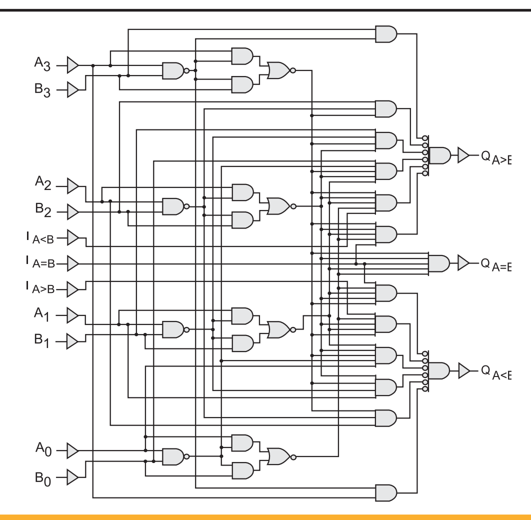
Activité
Réaliser l'activité **N°2** du **TP A2-1** dans le manuel d'activités : cette activité consiste à mettre en oeuvre un comparateur integré de la famille **XX85** et verifier son fonctionnement en fonction des niveaux logiques appliqués à ses entrées.

4-2 Extrait du document technique :
Extrait de la documentation technique du circuit comparateur : **74hct85**

FUNCTION TABLE

CAMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A_3, B_3	A_2, B_2	A_1, B_1	A_0, B_0	$I_{A>B}$	$I_{A<B}$	$I_{A=B}$	$Q_{A>B}$	$Q_{A<B}$	$Q_{A=B}$
$A_3 > B_3$	X	X	X	X	X	X	H	L	L
$A_3 < B_3$	X	X	X	X	X	X	L	H	L
$A_3 = B_3$	$A_2 > B_2$	X	X	X	X	X	H	L	L
$A_3 = B_3$	$A_2 < B_2$	X	X	X	X	X	L	H	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 > B_1$	X	X	X	X	H	L	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 < B_1$	X	X	X	X	L	H	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 > B_0$	X	X	X	H	L	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 < B_0$	X	X	X	L	H	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	H	L	L	H	L	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	L	H	L	L	H	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	L	L	H	L	L	H
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	X	X	H	L	L	H
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	H	H	L	L	L	L
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	L	L	L	H	H	L

Notes
1. H = HIGH voltage level
L = LOW voltage level
X = don't care

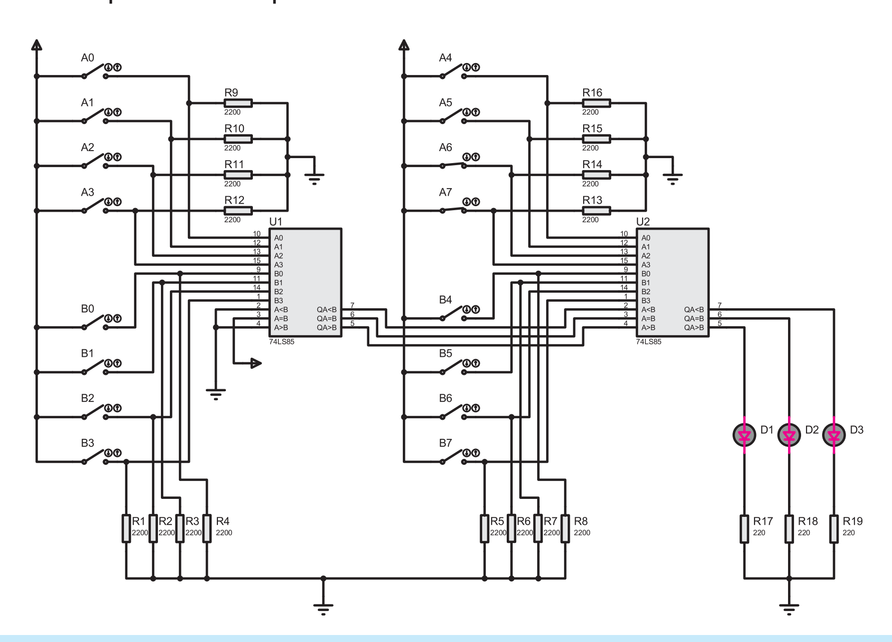




5- Comparaison de deux nombres de plus de 4 bits

A titre d'exemple, pour comparer deux nombres de **8 Bits** $A=a_7a_6a_5a_4a_3a_2a_1a_0$ et $B=b_7b_6b_5b_4b_3b_2b_1b_0$, on utilise deux circuits de type **XX85** montés en cascade comme suit :

- Relier les Broches " **5** ", " **6** " et " **7** " du comparateur gérant les 4 bits de poids faibles respectivement aux entrées " **4** ", " **3** " et " **2** " du comparateur gérant les 4 bits de poids forts.
- Les niveaux logiques " **0** ", " **1** " et " **0** " sont appliqués respectivement sur les broches " **4** ", " **3** " et " **2** " du premier comparateur.



Activité

Réaliser l'activité **N°3** du **TP A2-1** dans le manuel d'activités : cette activité consiste à mettre en cascade deux comparateurs integrés de la famille **XX85** en vue de comparer deux nombres binaires de 8 bits chacun.

C- RESUMÉ

Pour monter en cascade deux comparateurs, il faut que les sorties de celui de poids le plus faible soient connectées aux entrées ayant les mêmes indices dans le comparateur de rang supérieur :

QA>B ----------- A>B

QA<B ----------- A<B

QA=B ----------- A=B

Les comparateurs sont employés dans les circuits de décodage des adresses des ordinateurs ; ils sont aussi très utiles dans les applications de régulation où un nombre binaire décrivant le comportement d'une variable physique régulée (vitesse, position, etc..) est comparé à une valeur de consigne. Les sorties du comparateur dans ce cas servent de déclencheur à l'envoi de signaux pour le pilotage et la conduite des mécanismes qui ramènent la variable physique vers son point de consigne.

D- EXERCICES

Exercice 1 :
Soit à comparer les deux nombres suivants : **A(a_1a_0)** et **B(b_1b_0)** ; en ne tenant compte que du cas " **A égal B** ".
1- Etablir la table de vérité correspondante.
2- Déduire l'équation logique.
3- Etablir le logigramme correspondant.

Exercice 2 :
On désire réaliser un opérateur capable d'effectuez la comparaison de 2 nombres positifs **A** et **B** codés sur **4 bits**. La sortie **S** de l'opérateur vaut " **1** " si **A** est **strictement inférieur** à **B**, sinon, elle vaut " **0** "
- ✔ **S = 1 si A < B**,
- ✔ **S = 0** si **A > B** ou **A = B**.

1- Proposer une solution à l'aide d'un soustraite.
2- Une autre solution consiste à comparer bit à bit les nombres **A** et **B** en commençant par les bits de poids forts. L'algorithme utilisé est le suivant :
S = 1 si ($a_3 < b_3$)
OU (($a_3 = b_3$) ET ($a_2 < b_2$))
OU (($a_3 = b_3$) ET ($a_2 = b_2$) ET ($a_1 < b_1$))
OU (($a_3 = b_3$) ET ($a_2 = b_2$) ET ($a_1 = b_1$) ET ($a_0 < b_0$)) .
2-a Construire l'opérateur élémentaire à 2 entrées a_i et b_i dont les sorties I_i (**Inférieur**) et E_i (**égal**) vérifient :
$I_i = 1$ si $a_i < b_i$, $I_i = 0$ sinon.
$E_i = 1$ si $a_i = b_i$, $E_i = 0$ sinon.
2-b En utilisant l'opérateur réalisé précédemment, proposer le schéma complet du comparateur.

Exercice 3 :
En utilisant les cellules suivantes :
- Additionneur à **1bit**.
- Comparateur à **1bit**.
- Porte logique **"OU exclusif"**.

1- Pour **A** : a_0 et **B** : b_0, tracer le logigramme réalisant la fonction **"S"** suivante :
- Si **A ≥ B : S = A + B**
- Si **A < B : S = A - B**

2- Déduire le logiramme pour des nombres à 4 bits : **A : $a_3a_2a_1a_0$** et **B : $b_3b_2b_1b_0$**.

6- Problème de synthèse :
Reprenons le système "distributeur automatique de billets".
1- Citer tous les nombres décimaux qui peuvent être représentés sur deux variables binaires.
2- Pour que le client puisse saisir son code secret, de combien de touches doit être muni le clavier ? citer les chiffres possibles du code secret.
3- Dresser une table de vérité complète " sur **4 variables** " de la comparaison de ces nombres. Les résultats " **B sup A** ", " **B égal A** " et " **B inf A** " devront apparaître.
4- Avec une grille de Karnaugh, présenter le cas " **B égal A** " en positionnant les 4 variables " b_0 ", " b_1 ", " a_0 " et " a_1 ".
5- En déduire l'équation logique de " **B égal A** " ; transformer cette équation de manière à mettre en évidence l'emploi de portes **XOR** et **NOT**, fonction de " b_0, b_1, a_0, a_1 ".
6- Traduire cette équation en logigramme.

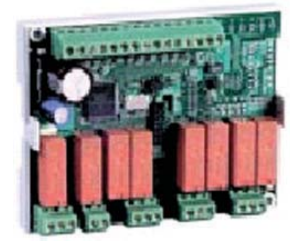
UNITÉ ARITHMÉTIQUE ET LOGIQUE " UAL "

A- MISE EN SITUATION

1- Fonctionnement du système : Serres agricoles



Dans les serres agricoles, une carte de contrôle logique permet de centraliser la gestion de l'arrosage automatique.
Les options de programmation offertes par cette carte permettent d'optimiser à la fois le volume d'eau consommé et le contrôle direct de l'ouverture des électro vannes (24 VDC).
Le circuit électronique de commande est construit autour d'un micro-contrôleur.



Les informations recueillies par les capteurs installés dans tous les coins des serres sont envoyées vers un sous-ensemble du microcontrôleur spécialisé dans le traitement arithmétique et logique des données.
Ce sous-ensemble est appelé ***Unité Arithmétique et Logique (UAL)***.

2- Problèmes posés :

a) Quelle est la structure électronique d'une UAL?
b) Comment sont traitées les informations logiques et arithmétiques ?
c) Quelle forme peut prendre une UAL?

B - ARCHITECTURE D'UNE UAL.

1- Présentation :

Le rôle de l'**unité arithmétique et logique** est de réaliser des opérations *logiques* et *arithmétiques* de base :

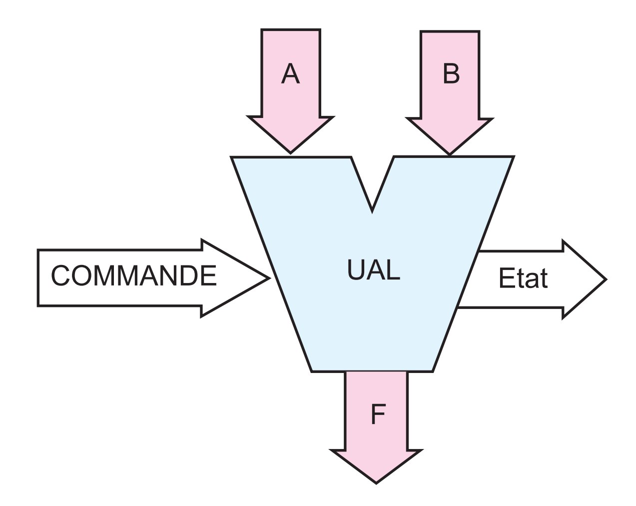
- Opérations *logiques* **bit** à **bit** " **ET**, **OU**, **NON**, **XOR**…".
- Opérations *arithmétiques* "**addition**, **soustraction**, **complémentations** à **1** ou à **2**..."
- Comparaisons
- Décalages et rotations.

2- Constitution :

L'**unité arithmétique et logique**, dispose de :

- Deux entrées *A* et *B* sur lesquelles on présente les données à traiter, *"les opérandes"*.
- Une sortie *F* donnant le résultat de l'opération effectuée.

- Une entrée *Commande* permettant le choix de l'opération à effectuer.
- Une sortie *Etat* donnant l'état de l'**UAL** après exécution de l'opération.



3- Constitution :

De part son nom, l'**UAL** peut être décomposée en deux grands blocs fictifs :
L'*Unité Logique* (**UL**), qui permet d'effectuer les opérations logiques **ET**(**AND**), **OU**(**OR**), **NON**(**NOT**), **OU exclusif**(**XOR**), etc... sur deux nombres de n bits placés en entrées. Ces opérations sont effectuées **bit** à **bit**.
Exemple : Opérations Logiques sur deux nombres de **4 bits**

- Complémentation NOT (1011) = (0100)
- Produit logique (1101) ET (1011) = (1001)
- Somme logique (1101) OU (1011) = (1111)
- Somme exclusive logique (1101) Å (1011) = (0110)

L'Unité Arithmétique (**UA**), qui permet d'effectuer des opérations arithmétiques, telle que l'addition et la soustraction sur 2 nombres de n bits placés en entrée.
Exemple : Opérations arithmétiques sur deux nombres de **4 bits**.

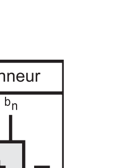
Addition	(1101) + (1011) = 1(1000) *1 est la retenue (carry)*
Soustraction	(1101) – (1011) = (0010)

C - REALISATION DES UALs :

Les **UALs** existent, sous forme de circuits intégrés indépendants, ou intégrées dans d'autres circuits numériques spécialisés tels que les microprocesseurs ou les microcontrôleurs.

1- Principe :

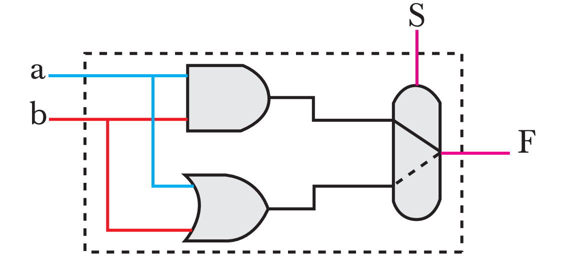
Pour construire une **UAL** simple, on peut combiner ensemble les opérateurs entre eux.



Opérateur NOT	Opérateur ET	Opérateur OU	Opérateur XOR	Multiplexeur	Additionneur
					a_n b_n R_{n-1} + S_n R_n
(1)	(2)	(3)	(4)	(5)	(6)

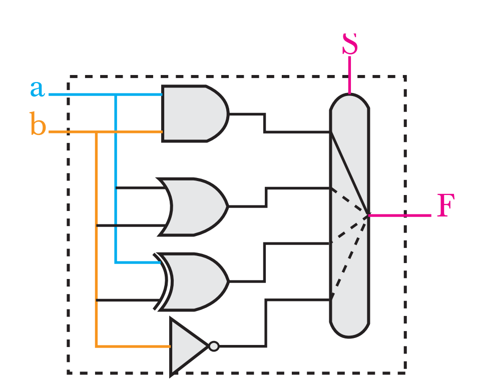
2- UL 1 bit réalisant les opérations AND et OR :

L'association des trois blocs **(2)**, **(3)** et **(5)** de la figure 4 permet de réaliser une **unité logique à 1 seul bit**, dans laquelle les bits **a** et **b** sont les entrées, **S** le bit de commande et **F** la sortie.
Le multiplexeur sélectionne **a ET b** ou **a OU b** selon que **S** vaut **0** ou **1**.



3- UL 1 bit réalisant les opérations AND, OR, NOT et XOR :

L'association des blocs **(1)**, **(2)**, **(3)**, **(4)** et **(5)** permet d'obtenir la configuration suivante :



S	F
00	a ET b
01	a OU b
10	a XOR b
11	NOT (b)

4- UA 1 bit " additionneur-soustracteur " :

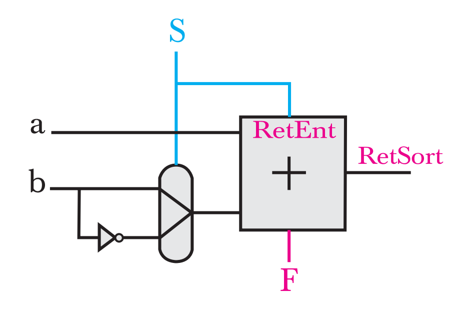
L'association des blocs **(1)**, **(5)** et **(6)** permet de réaliser une **unité arithmétique 1 bit.**

Si S = 0 :

$F = a + b + RetEnt = a + b + 0 = a + b$

Si S = 1 :

$F = a + \overline{b} + RetEnt = a + (\overline{b} + 1) = a - b$

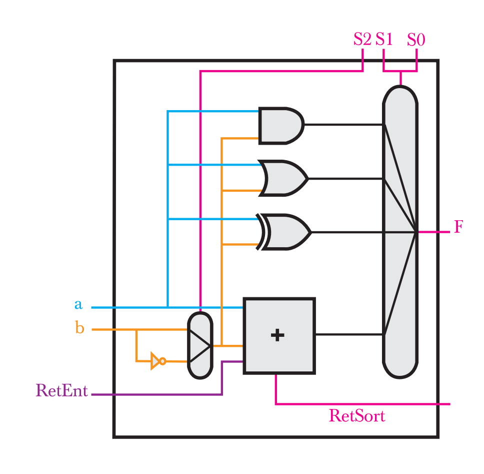


Activité
Réaliser l'activité **N°1** du **TP A2-2** dans le manuel d'activités : cette activité consiste à verifier le fonctionnement d'une UL élémentaire réalisée à base de circuits logiques.

5- UAL 1 bit :

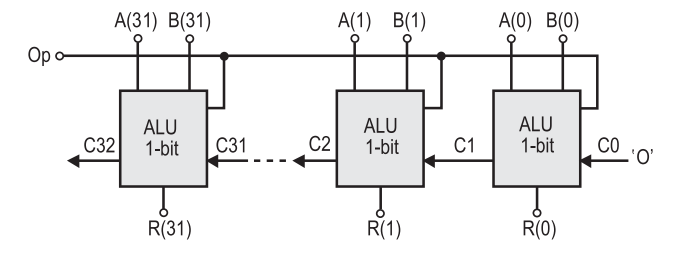
L'association des 6 blocs permet de réaliser une **unité arithmétique** et **logique** 1 bit.

S			RetENt	F
S_2	S_1	S_D		
0	0	0	x	a ET b
0	0	1	x	a OU b
0	1	0	x	a XOR b
0	1	1	0	a + b
0	1	1	1	a + b + 1
1	0	0	x	a ET NOT (b)
1	0	1	x	a OU NOT (b)
1	1	0	x	a XOR NOT (b)
1	1	1	0	a – b – 1
1	1	1	1	a – b



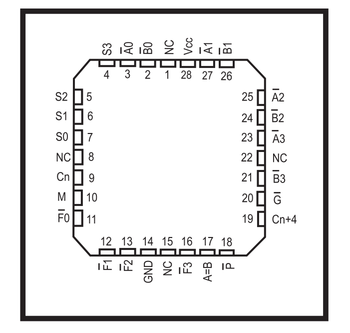
6- UAL n bits :

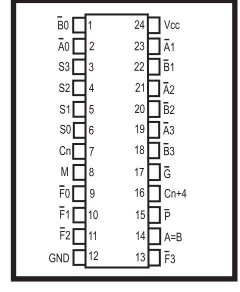
Pour réaliser une **UAL** n bits, on associe **n UALs** 1 bit en cascade :
Exemple : **UAL 32 bits**



D- L'UAL EN CIRCUITS SPECIALISES

Parmi les circuits spécialisés on cite en **TTL** : **74LS181 / 74LS381** et le **74LS382**. L'étude portera sur le **74LS181** ce circuit est une *UAL* de **4 bits**. Il est destiné à être connecté en cascade avec d'autres **74LS181** pour traiter des données de taille supérieure à **4 bits.**





1- Les entrées du circuit 74LS181 :

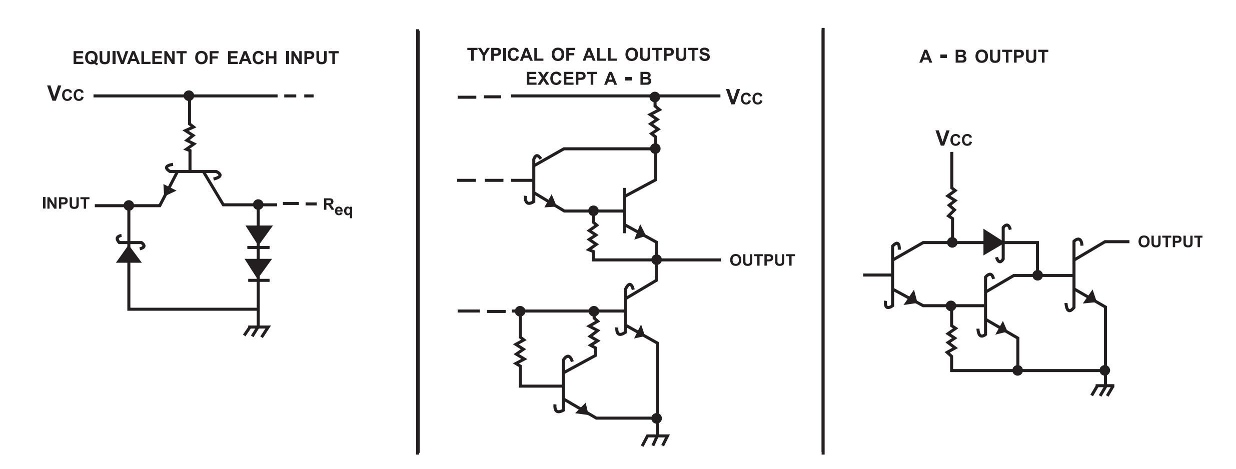
Ce circuit ayant entre autres fonctions, celle d'additionner ou de soustraire, il est doté de 4 entrées pour le nombre A et 4 autres pour le nombre B. Ce sont les entreés :

$$\overline{A_3}\,\overline{A_2}\,\overline{A_1}\,\overline{A_0} \qquad \overline{B_3}\,\overline{B_2}\,\overline{B_1}\,\overline{B_0}$$

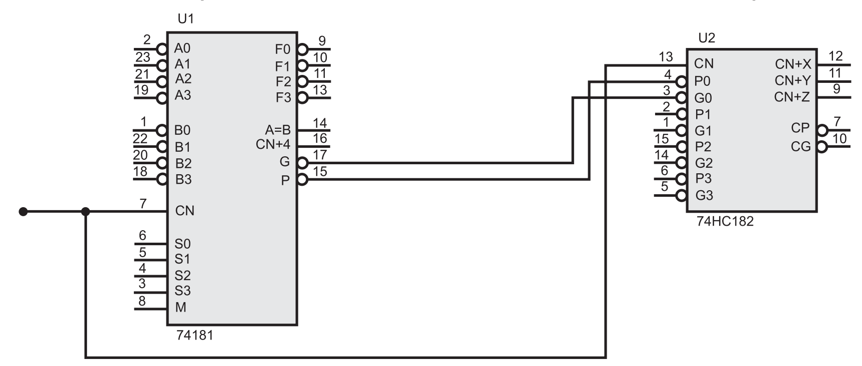
Entre outre, nous avons une entrée pour la retenue de poids faible appelée C_n.
Cette entrée doit etre à “0” pour l’addition et à “1” pour la soustraction (voir cours sur les opérations arithmétiques).

2- Les sorties du circuit 74LS181:

Dans ce cadre, on cite :
- les 4 sorties intitulées : $F_3F_2F_1F_0$ elles correspondent aux quatre bits fournissant le résultat des différentes opérations.
- La retenue de poids fort récupérée sur la broche C_{n+4}.
- Une sortie **A = B** indiquant l’égalité des deux nombres **A** et **B** (cette sortie est à collecteur ouvert)



- Deux sorties **X** et **Y** destinées à la liaison avec un circuit de la famille **74xx182** (générateur anticipant des retenues pour des circuits **74xx181**, ce circuit n’est pas à developper).



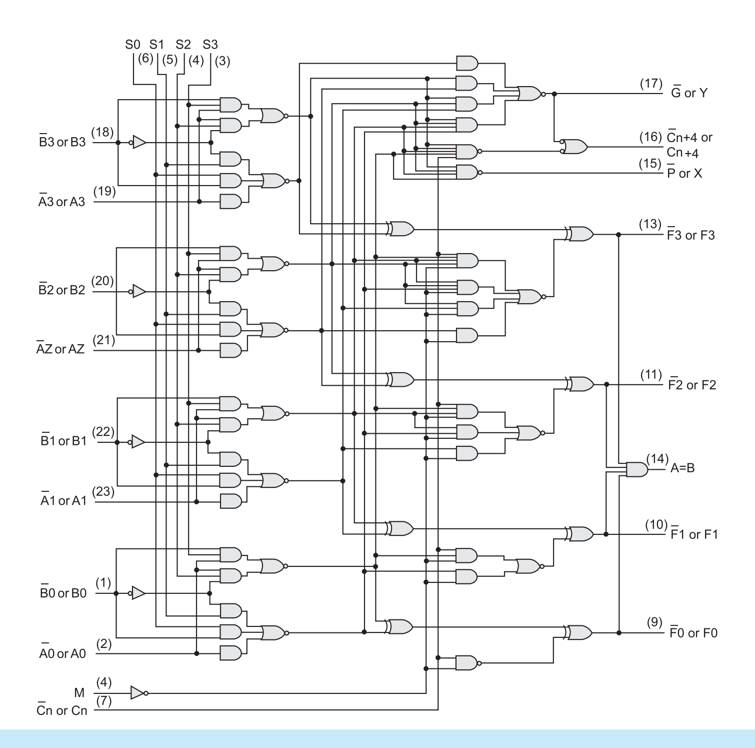
3- Les commandes du 74LS181 :

La commande **M** à l’état “Haut” permet de réaliser les fonctions logiques.
M à l’état “Bas” transforme le circuit en calculateur arithmétique (avec quelques fonctions logiques).
Quatre entrées de sélection : $S_3S_2S_1S_0$, permettent dans chaque cas (**M=0** ou **M=1**) 16 combinaisons possibles, ce qui permet de réaliser 32 fonctions différentes.
La table ci-dessous, extraite des documents du constructeur, énumère ces différentes fonctions.

1- Table de fonctionnement :

SELECTION				ACTIVE LOW DATA		
				M - H LOGIC FUNCTIONS	M-L : ARITHMETIC OPERATIONS	
S_3	S_2	S_1	S_0		Cn – L (ms carry)	Cn – M (with carry)
L	L	L	L	F = $\overline{A}$	F = A MINUS 1	F = A
L	L	L	H	F = $\overline{AB}$	F = AB MINUS 1	F = AB
L	L	H	L	F = $\overline{A}$ + B	F = AB MINUS 1	F = A$\overline{B}$
L	L	H	H	F = 1	F = MINUS 1 (2+COMP)	F = ZERO
L	H	L	L	F = $\overline{A + B}$	F = A PLUS (A + $\overline{B}$)	F = A PLUS (A + $\overline{B}$) PLUS 1
L	H	L	H	F = $\overline{B}$	F = AB PLUS (A + $\overline{B}$)	F = AB PLUS (A + $\overline{B}$) PLUS 1
L	H	H	L	F = $\overline{A \oplus B}$	F = A MINUS B MINUS 1	F = A MINUS B
L	H	H	H	F = A + $\overline{B}$	F = A + $\overline{B}$	F = (A + $\overline{B}$) PLUS 1
H	L	L	L	F = $\overline{A}$B	F = A PLUS (A + B)	F = A PLUS (A + B) PLUS 1
H	L	L	H	F = A ⊕ B	F = A PLUS B	F = A PLUS B PLUS 1
H	L	H	L	F = B	F = AB PLUS (A + B)	F = A$\overline{B}$ PLUS (A + B) PLUS 1
H	L	H	H	F = A • B	F = (A + B)	F = (A + B) PLUS 1
H	H	L	L	F = 0	F = A PLUS A	F = A PLUS A PLUS 1
H	H	L	H	F = A$\overline{B}$	F = A$\overline{B}$ PLUS A	F = AB PLUS A PLUS 1
H	H	H	L	F = AB	F = AB PLUS A	F = A$\overline{B}$ PLUS A PLUS 1
H	H	H	H	F = A	F = A	F = A PLUS 1

2- Logigramme :



Activité

Réaliser l'activité **N°2** du **TP A2-2** dans le manuel d'activités : cette activité consiste à mettre en oeuvre une **U**nité **A**rithmétique et **L**ogique de la série **74XX18X.**



E - RESUMÉ

Dans les machines ou les systèmes numériques, les opérations arithmétiques et logiques sont confiées à l'unité arithmétique et logique (UAL), celle-ci est composée de portes logiques, de bascules, de circuits arithmétiques combinés de manière à pouvoir effectuer des additions, des soustractions, des comparaisons, etc ... de nombres binaires.

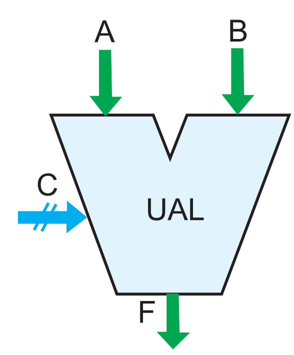
De nombreux microprocesseurs ne disposent pas dans leur unité arithmétique et logique des circuits (matériel) en mesure d'effectuer des opérations complexes (multiplication, division, etc…). Les machines numériques basées sur de tels microprocesseurs, accomplissent ces opérations au moyen d'un programme (logiciel). Le recours à de tels logiciels, se traduit par moins de circuits dans le microprocesseur mais en contre partie par un temps de calcul plus long.

F - EXERCICES

1- Exercice résolu :

Soit le circuit de la figure ci-contre, représentant une Unité Logique élémentaire.
A et **B** sont des entrées sur **1bit**.
La commande **C** est sur **2bits**.
Elle permet de sélectionner une opération parmi les 4 fonctions que peut effectuer l'unité.
Faire le schéma de l'unité en utilisant :

- a- Uniquement des portes logiques.
- b- Des multiplexeurs et des portes logiques.



Sélection		Fonction
C_1	C_0	
0	0	S=A.B
0	1	$S=A.\overline{B}$
1	0	S=A.B
1	1	$S=A.\overline{B}$

Solution

1- D'après la table de fonctionnement on peut écrire :

$$F = \overline{C_0}.\overline{C_1}.(A.B) + C_0.\overline{C_1}.(A.\overline{B}) + \overline{C_0}.C_1.(A + B) + C_0.C_1.(A + \overline{B})$$

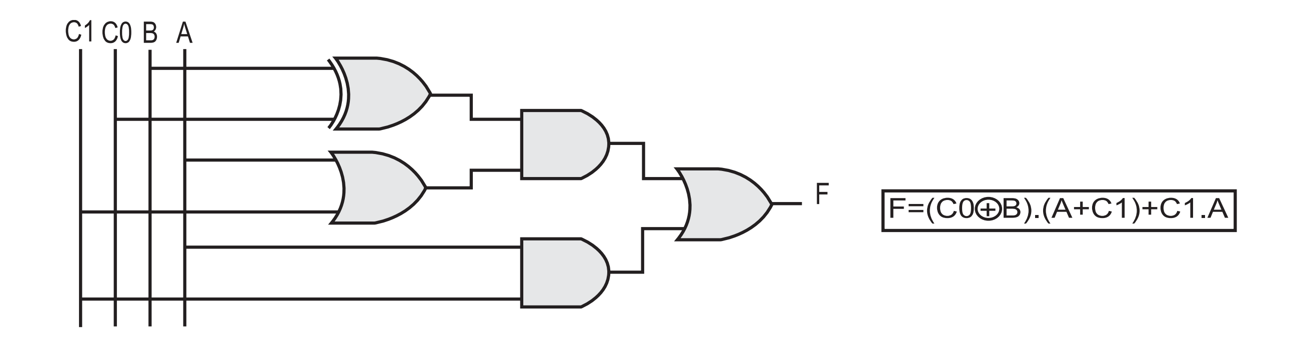
$$F = \overline{C_1}.A.(\overline{C_0}.B + C_0.\overline{B}) + C_1(\overline{C_0}.(A + B) + C_0.(A + \overline{B}))$$

$$F = \overline{C_1}.A.(C_0 \oplus B) + C_1(\overline{C_0}.A + \overline{C_0}.B + C_0.A + C_0.\overline{B})$$

$$F = \overline{C_1}.A.(C_0 \oplus B) + C_1.(A + (C_0 \oplus B))$$

$$F = \overline{C_1}.A.(C_0 \oplus B) + C_1.A + A(C_0 \oplus B)$$

$$F = (C_0 \oplus B)(\overline{C_1}.A + A) + C_1.A$$

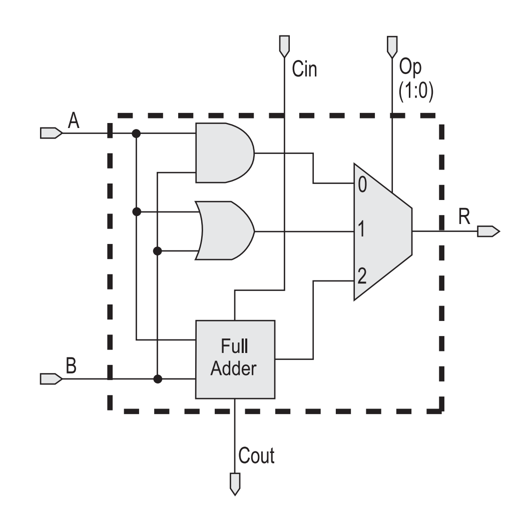


2- On peut utiliser deux multiplexeurs l'un pour sélectionner la fonction **ET** ou **OU** et l'autre pour sélectionner **B** ou son complément.

2- Exercices à résoudre :

Exercice 1 :

Le schéma ci-contre, symbolise une **UAL-1bit** élémentaire.



1- Quelles sont les opérations arithmétiques réalisées par ce circuit ?

2- Quelles sont les opérations logiques réalisées par ce circuit ?

3- Dresser la table de fonctionnement relative à ce circuit.

Exercice 2 :

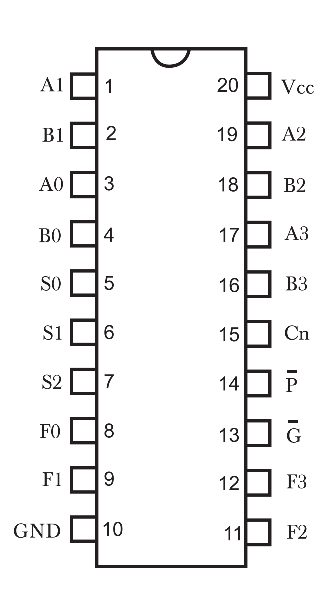
Le schéma suivant, symbolise une **UAL-1bit** élémentaire :

Le circuit intégré **74LS381** est une **UAL** qui permet de réaliser des opérations logiques ou arithmétiques entre deux mots de **4 bits** :

le mot **A = $A_3A_2A_1A_0$** et le mot **B = $B_3B_2B_1B_0$**

Sa table de fonctionnement est la suivante :

S_2	S_1	S_0	Opérations réalisées
0	0	0	F = 0000
0	0	1	F = B – A
0	1	0	F = A – B
0	1	1	F = A plus B
1	0	0	F = A OU B
1	0	1	F = A XOR B
1	1	0	F = A ET B
1	1	1	F = 1111



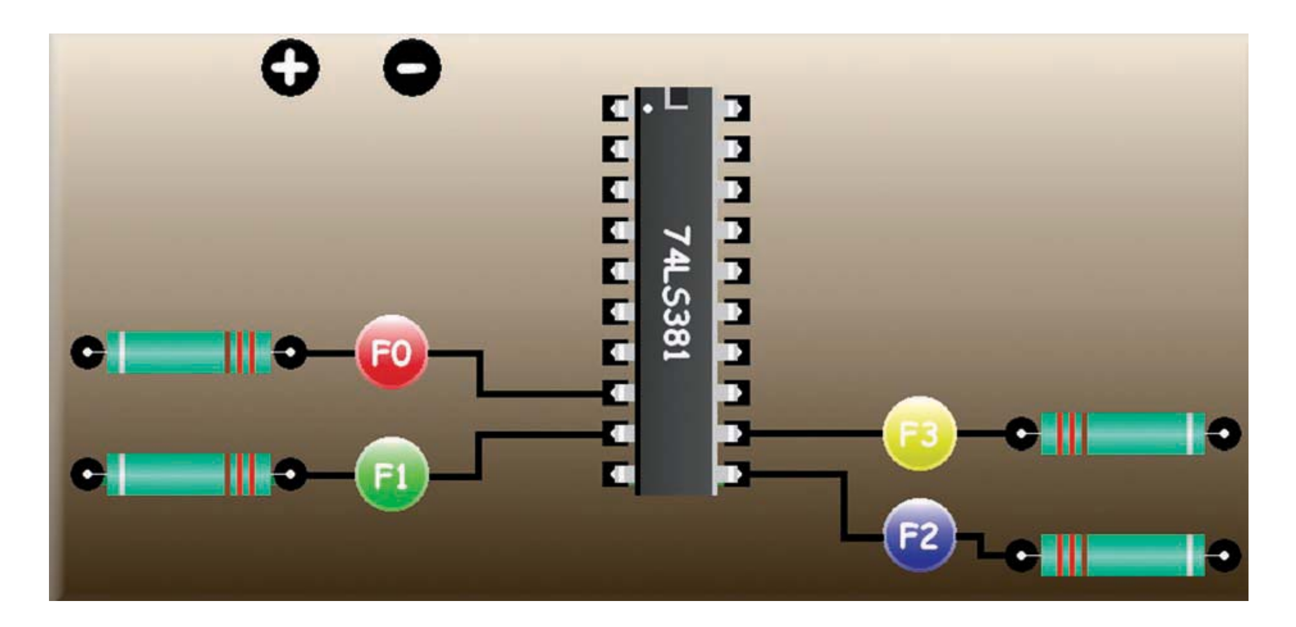
1- Quelles sont les opérations arithmétiques réalisées par ce circuit ?
2- Quelles sont les opérations logiques réalisées par ce circuit ?
3- Sur le cahier de cours, reproduire puis compléter la table ci-dessous.

$S(S_2S_1S_0)$	$A(A_3A_2A_1A_0)$	$B(B_3B_2B_1B_0)$	$F(F_3F_2F_1F_0)$
011	0101	0110	....
	1010	0111	....
110	0111	0101	....
	1011	0101	....
....	0110	1100	1010
....	1100	1010	1110

a/ Sur le cahier de cours, reproduire puis compléter le câblage de circuit ci-aprés pour avoir à la sortie :

F(F3F2F1F0)= A(1000) OU B(1010)

b/ Indiquer les états des Leds “allumées ou éteintes”.





Exercice 3 : Soit le logigramme suivant :

1- Pour chacune des positions du commutateur “**SW**”, dites quel est l’état de chacune des sorties suivantes : **S1**, **S2**, **S3**, **S4**, **S** et **E**.

NB : Présenter les résultats dans un tableau.

2- Déduire les équations de ces sorties.

3- Quel est le rôle du Bloc “**ADD**” et quel nom peut-on lui accordé ?

4- Quel est le rôle des résistances R_1 à R_4 ?

LES COMPTEURS

A- MISE EN SITUATION

1- Présentation du système technique : La Compteuse industrielle



Les gélules ou comprimés par exemple, sont soumis à l'action de la force centrifuge et s'alignent progressivement le long du bord de la cuve sans être émiettés, effritées ou endommagées.
Dans une gamme étendue, le passage d'un produit à un autre s'effectue en 2 minutes par des réglages mécaniques ainsi que l'appel d'un programme mémorisé dans l'armoire électronique.



En standard, les paramètres de 20 produits sont mémorisés. L'opérateur peut alors afficher, à l'aide du clavier et de l'afficheur LCD les consignes d'exploitation telles que le nombre de lots, nombre de pièces par lot,...
En permanence, l'opérateur visualise en clair les informations sur le travail en cours. Ceci permet, par exemple l'intégration des compteuses pharmacie dans un laboratoire ou une chaîne automatique de production.
La vitesse de comptage est très élevée car la fréquence de passage de comprimés ou gellules est importante. Le compteur asynchrone est incapable de suivre le rythme imposée par la fabrication car la mise en cascade de ses différentes bascules crée un retard qui peut dépasser l'unité de production et engendrer des erreurs de comptage.

2- Problèmes posés :

Déterminer les caractéristiques d'un compteur synchrone :
- **+ la table de comptage**
- **+ le mode de cablage des bascules**

Trouver le ou les processus d'initialisation :
- **+ processus cablé**
- **+ processus prépositionné.**



B- RAPPEL :

Définition d'un compteur :

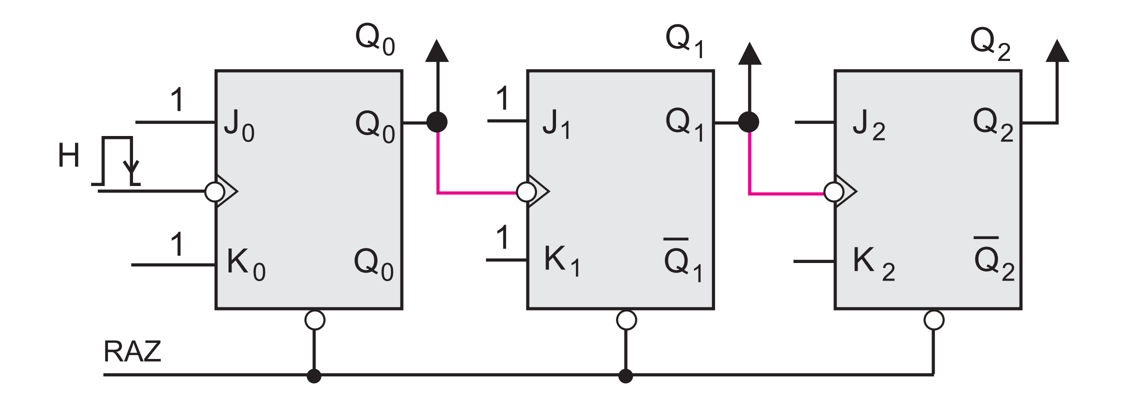
La fonction qui s'impose pour dénombrer des événements numériques est le compteur.
Un compteur résulte de l'assemblage d'un ensemble combinatoire et séquentiel cadencé par un signal d'horloge **H** matérialisant les événements à compter ou à dénombrer.

Capacité d'un compteur (modulo) :

Un compteur modulo **N** (diviseur de fréquence par **N**), comporte **N** combinaisons de **0** à **(N-1)** et utilise n bascules tel que : $2^{n-1} < N < 2^n$

Compteur Asynchrone (en binaire naturel) :

La réalisation d'un compteur asynchrone consiste à mettre en cascade des bascules. Un compteur asynchrone reçoit le signal d'horloge seulement sur l'entrée **H** de la première bascule. Pour les autres bascules l'entrée d'horloge de chaque bascule est reliée à la sortie de la bascule précédente de la façon suivante :



Avantages :

Conception facile par répétition de modules diviseurs de fréquence par deux.

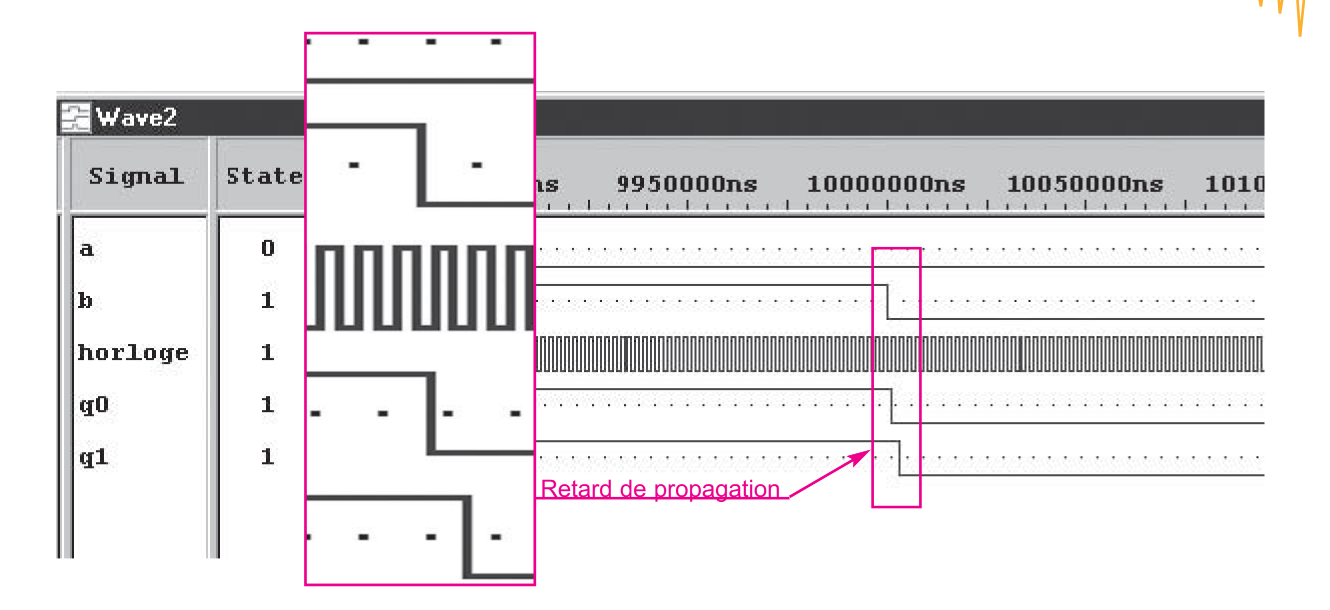
D reliée à **Q** pour une bascule **D**

ou

J=K=1 pour une bascule **J-K**

Inconvénients :

* Temps d'établissement long
* Passage par des états de sortie parasites
  Commutation en cascade des bascules.



Le chronogramme ci-dessus montre bien le retard de la propagation de l'information de la première bascule à la deuxième.

C- LES COMPTEURS/DÉCOMPTEURS SYNCHRONES :

Activité

Réaliser l'activité N°1 du TP3 dans le manuel d'activités

Pour étudier le fonctionnement du compteur de gélules, on utilise une maquette d'essai qui permettra de simuler l'opération de comptage. On demande de remplir la table de comptage relative au compteur utilisé sur cette maquette et d'en déduire le cycle de fonctionnement.

1- Synthèse du fonctionnement d'un compteur/décompteur synchrone

Cette synthèse consiste à :

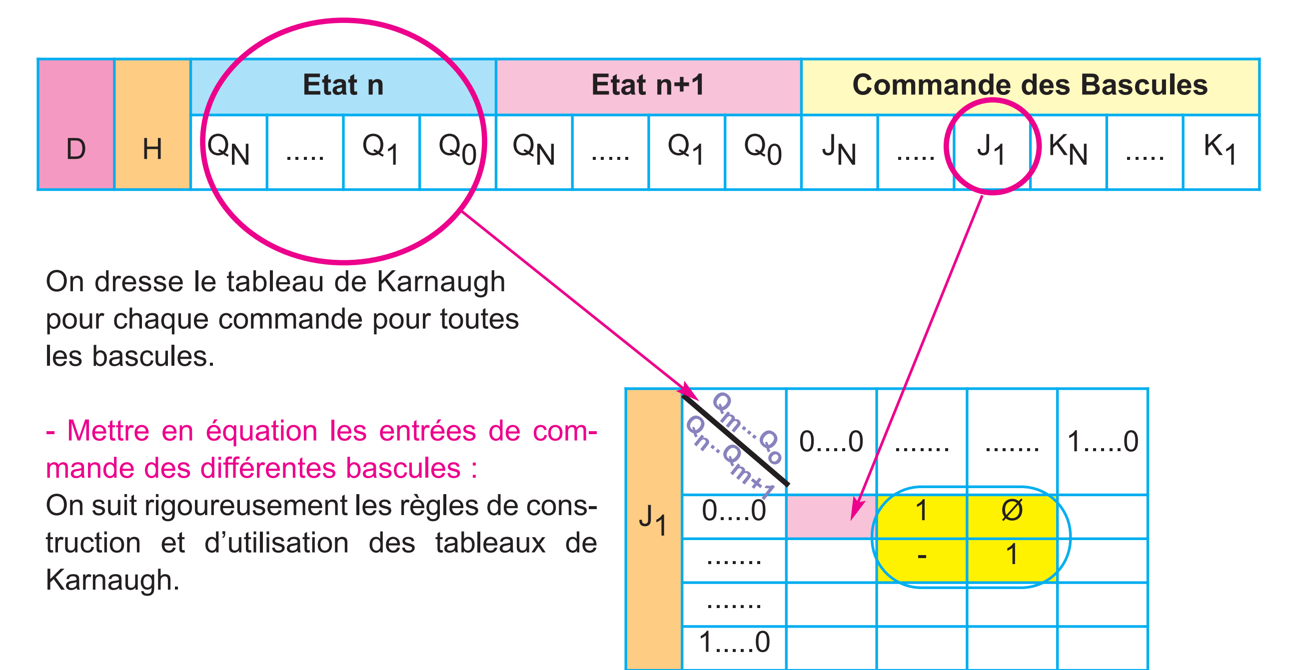
- Etablir la table de comptage :

C'est un tableau qui résume avec précision les étapes du cycle de fonctionnement. Elle est établie après le choix du nombre de bascules qui constituent le compteur et de leur type. Elle traduit les états des sorties des bascules avant **(état n)** et après **(état n+1)** l'impulsion d'horloge.

		Etat n				Etat n+1				Commande des Bascules					
D	H	Q_N	.....	Q_1	Q_0	Q_N	.....	Q_1	Q_0	J_N	.....	J_1	K_N	.....	K_1

A partir du diagramme de fluence relatif à la bascule choisie et de son comportement dans la table de comptage on détermine les états logiques de ses différentes entrées de commande.

- Etablir la table de fonctionnement des différentes bascules :

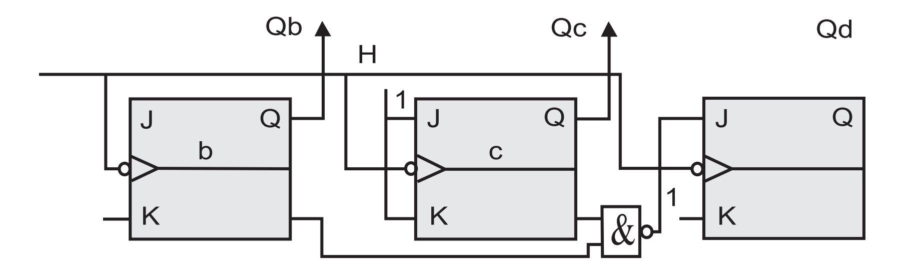


On dresse le tableau de Karnaugh pour chaque commande pour toutes les bascules.

- Mettre en équation les entrées de commande des différentes bascules :

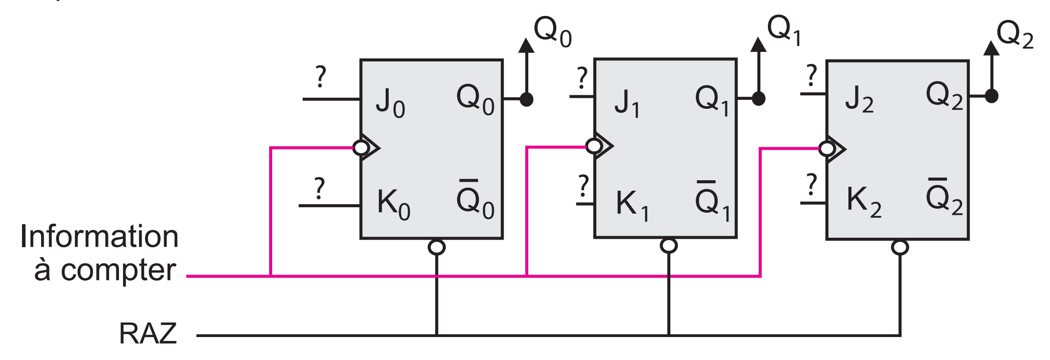
On suit rigoureusement les règles de construction et d'utilisation des tableaux de Karnaugh.

- Tracer le schéma de câblage.



2- Spécificités d'un compteur synchrone :

Dans un compteur synchrone l'horloge est commune à toutes les bascules. Il y a donc un seul temps de propagation pour obtenir le résultat du comptage. Les états parasites "GLITCH" n'existent pas.





Activité
Réaliser l'activité N°2 du TP3 dans le manuel d'activités
Il s'agit de réaliser (partiellement) sur plaque à essai un compteur synchrone à base de bascules JK conformément aux équations trouvées dans l'activité 1.

Avantages :

* Temps d'établissement constant (Une unité de propagation)
* Pas de valeurs parasites.

Inconvénients :

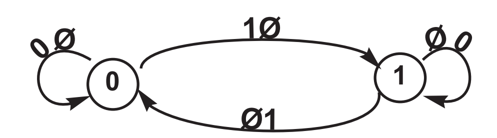
* Utilise des circuits logiques combinatoires en plus des bascules.
* Les modules sont donc différents

3- Exemple d'application :

On désire concevoir un décompteur synchrône des nombres impairs dont le cycle de décomptage est le suivant : 7 - 5 - 3 - 1.

Solution :

a- Nombre de bascules nécessaires N=3 (2^3 = 8 nombre supérieur à 7)
b- Type de bascules choisies: JK (la résolution sera différente si on choisit D ou T)
c- Rappel du diagramme de fluence de la bascule JK

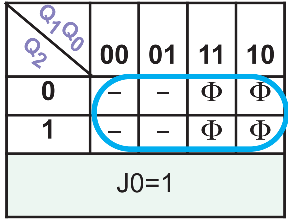


d- Table de décomptage :

D	H	Etat n			Etat n+1			Commande des Bascules					
		Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	J_2	J_1	J_0	K_2	K_1	K_0
7	↓	1	1	1	1	0	1	μ_1	δ	μ_1	μ_1	δ	μ_1
5	↓	1	0	1	0	1	1	δ	ε	μ_1	δ	ε	μ_1
3	↓	0	1	1	0	0	1	μ_0	δ	μ_1	μ_0	δ	μ_1
1	↓	0	0	1	1	1	1	ε	ε	μ_1	ε	ε	μ_1



e- Mise en équation des entrées de commande :

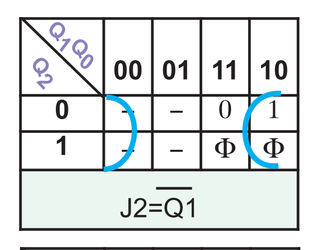


Q_2 \ Q_1Q_0	00	01	11	10
0	–	–	Φ	Φ
1	–	–	Φ	Φ

J0=1

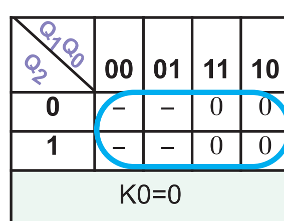
Q_2 \ Q_1Q_0	00	01	11	10
0	–	–	1	Φ
1	–	–	1	Φ

J1=1



Q_2 \ Q_1Q_0	00	01	11	10
0	–	–	0	1
1	–	–	Φ	Φ

J2=$\overline{Q1}$

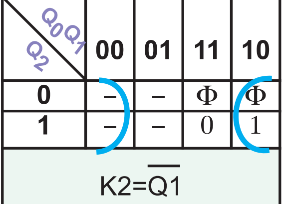


Q_2 \ Q_1Q_0	00	01	11	10
0	–	–	0	0
1	–	–	0	0

K0=0

Q_2 \ Q_1Q_0	00	01	11	10
0	–	–	1	Φ
1	–	–	1	Φ

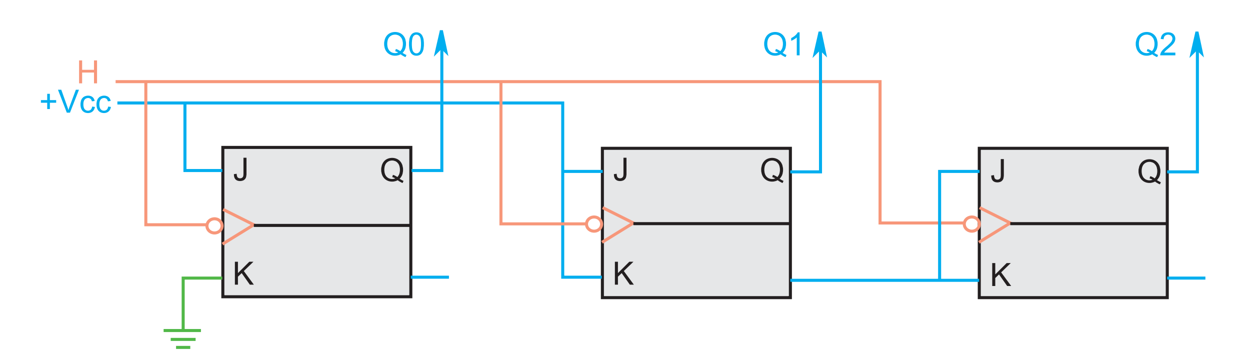
K1=1



Q_2 \ Q_0Q_1	00	01	11	10
0	–	–	Φ	Φ
1	–	–	0	1

K2=$\overline{Q1}$

f- Logigramme :



D- TECHNOLOGIE DES COMPTEURS INTÉGRÉS

Activité

Réaliser l'activité N°3 du TP3 dans le manuel d'activités

Il s'agit d'étudier un dispositif qui permet de gérer les défauts de comptage des gélules dans la compteuse industrielle. A cet effet on se propose de simuler sur ORCAD le fonctionnement d'un compteur/décompteur prépositionnable et de tracer le cycle de comptage pour différentes combinaisons des entrées de prépositionnement.

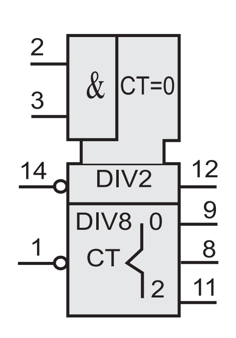
La plupart des compteurs intégrés, qu'ils soient synchrones (la majorité) ou asynchrones présentent des fonctionnalités diverses telles que :

- la remise à zéro.
- la mise à pleine capacité.
- le pré-chargement parallèle (ou prépositionnement parallèle).
- le choix entre comptage et décomptage.
- et bien d'autres encore.

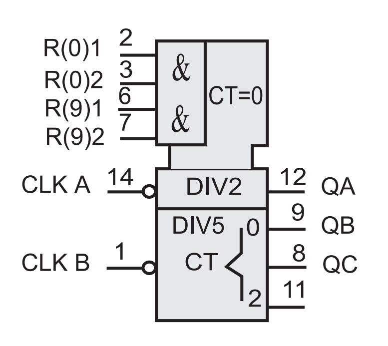
L'exploitation de la documentation technique fournie par le constructeur permet de retrouver les conditions de réalisations de différentes fonctionnalités à travers des tables de fonctionnement, des tables de vérité et des chronogrammes.

Exemples :

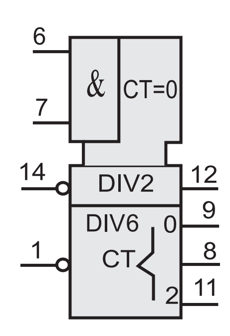
Compteur binaire 7493



Compteur décimal 7490



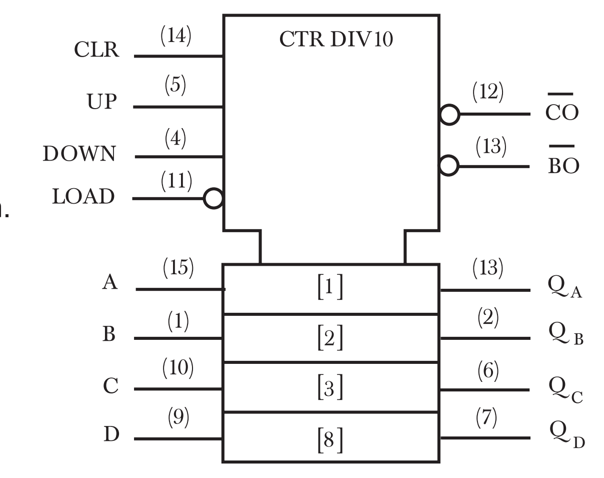
Compteur hex adécimal 7492



Le changement d'un compteur consiste à installer dans ce dernier un nombre de départ **p**. Ce nombre **p** est écrit sur les entrées de programmation puis il est chargé dans le compteur lorsque l'entrée **load** est active à 0 ou à 1, cela dépendra des compteurs.

Exemples : 74192

CLR : (CLEAR) : Entrée d'effacement(RAZ)
UP : Horloge du compteur.
DOWN : Horloge du décompteur.
/ LOAD : Entrée de chargement
A, B, C, D : Entrées de programmation.
QA, QB, QC, QD : Sorties du compteur ou décompteur
/ CO : Fin de comptage.
/ BO : Fin de décomptage.



E- RESUMÉ :

Définition d’un compteur :
Un compteur résulte de l'assemblage d'un ensemble combinatoire et séquentiel cadencé par un signal d'horloge **H** (information à compter).

Inconvénients du compteur asynchrone :
* Temps d'établissement long
* Passage par des états de sortie parasites

Avantages du compteur synchrone :
* Temps d'établissement constant (Une unité de propagation)
* Pas de valeurs parasites.

Inconvénients du compteur synchrone :
* Utilise des circuits logiques combinatoires en plus des bascules.
* Les modules sont donc différents

Système d’étude :

Compteuse industrielle :
Le compteur synchrone arrive, avec ses différents étages reliés tous à l’horloge, à suivre la cadence rapide de passage des gélules.

D- EXERCICE D'APPLICATION :

1- Exercice résolu :

Pour conditionner les gélules par plaque de 4, on installe un compteur décompteur sychrone réversible modulo **4**. Le mode de fonctionnement est donnée par la table de comptage ci-contre (**a** : commutateur à **2** positions "commutateur de sélection") :

$2^1 < 4 < 2^2$ donc il faut **2** bascules **D**

Compteur modulo **4** pour **a = 0**

décompteur modulo **4** pour **a = 1**

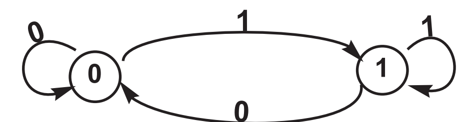


Table de comptage et de décomptage					
Etat (n)		Etat (n+1)			
		a = 0		a = 1	
Q_1	Q_0	Q_1	Q_0	Q_1	Q_0
0	0	0	1	1	1
0	1	1	0	0	0
1	0	1	1	0	1
1	1	0	0	1	0

* Tables de fonctionnement des différentes bascules

a \ Q_1Q_0	00	01	11	10
0	μ_0	ε	δ	μ_1
1	ε	μ_0	μ_1	δ

a \ Q_1Q_0	00	01	11	10
0	ε	δ	δ	ε
1	ε	δ	δ	ε

* Equation des entrées D_i

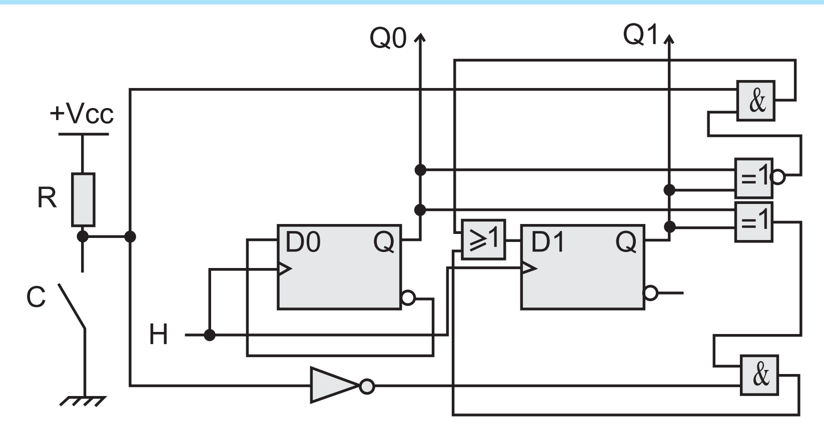
a \ Q_1Q_0	00	01	11	10
0	0	1	0	1
1	1	0	1	0

D_1

a \ Q_1Q_0	00	01	11	10
0	1	0	0	1
1	1	0	0	1

D_0

$D1 = \overline{a}.(\overline{Q1}.Q0+Q1.\overline{Q0})+ a.(\overline{Q1}.\overline{Q0}+Q1.Q0)$ $\quad$ $D0 = \overline{Q0}$



2- Exercice à résoudre :

Le conditionnement des gélules est fait en doses complètes, suffisantes pour un traitement de 15 jours, à raison de 3 gélules par jour. Le compteur utilisé pour réaliser cette opération est un 74190 ou 74191 dont les documents techniques du constructeur sont donnés ci-joint.

LOAD
DATA INPUTS: A, B, C, D
CLOCK
D/U
CTEN
QA
QB
QC
QD
MAX/MIN
RCD
13 14 15 0 1 2 2 2 1 0 15 14 13
LOAD — COUNT UP — INHIDIT — COUNT DOWN

1- Calculer le nombre de gélules nécessaire pour le traitement.

2- Proposer un schéma de cablage du circuit de comptage en utilisant le capteur optique H comme entrée à compter dans les cas où :

– le nombre total de gélules est à placer dans un flacon unique.

– le conditionnement est assuré sous forme de tablettes de 15 gélules chacune.

LES AUTOMATES PROGRAMMABLES INDUSTRIELS

A- MISE EN SITUATION :

1- Présentation du système technique :

Un système d'aide au stationnement des bus lors des arrêts.



Il s'agit d'un système d'aide à **l'accostage** et au guidage de bus qui permet au chauffeur de se garer au plus près du trottoir, dans des **stations courbes ou encaissées**, afin de faciliter l'accès des bus aux personnes fragilisées. **Dix capteurs photoélectriques**, placés sous le châssis, renvoient **les indications au chauffeur** sur son positionnement **par rapport à la ligne blanche.**

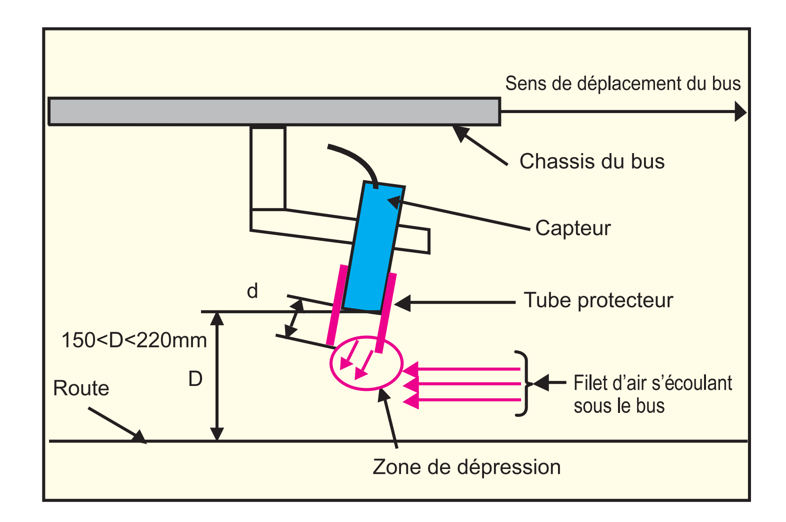


Un fonctionnement basé sur l'utilisation de détecteurs photoélectriques :



L'ensemble des tracés au sol constitue le point de démarrage du système d'aide. Il regroupe deux bandes blanches placées perpendiculairement à l'axe routier, dites bandes de "start", une bande blanche longitudinale parallèle à la bordure du trottoir pour le suivi de la trajectoire et enfin des pointillés, situés parallèlement à l'extrémité de la bande précédente, nécessaires à la détermination du point d'arrêt et de son codage en fin d'accostage.

Les tracés au sol sont détectés par un ensemble de 10 capteurs fixés sous le bus à proximité de l'essieu avant; 8 capteurs sont affectés au suivi de la bande longitudinale et 2 à la détection du type d'arrêt.
Le principe utilisé est basé sur les propriétés réfléchissantes des objets.





Pour limiter les projections de boue et autres impuretés sur la face sensible du détecteur, un tube protecteur est mis en place sur chaque capteur. Le mouvement d'air provoqué par le déplacement du bus permet de créer un effet venturi qui éjecte les impuretés.
La distance "d" est réglée pour obtenir l'effet optimal sans gêner l'émission ou la réception.

Le traitement des informations :

Le dialogue avec le chauffeur pour l'aide à l'arrêt longitudinal est réalisé par une rampe lumineuse verticale comportant 5 voyants. Le marquage au sol, détecté par le capteur d'arrêt longitudinal en combinaison avec les capteurs d'accostage latéral, est constitué de 3 bandes blanches. Pendant la phase finale d'acccostage, chaque apparition et disparition d'une de ces 3 bandes allume un voyant de la rampe lumineuse verticale. La dispartion de la deuxième bande active un signal sonore.

Ecran de dialogue



2- Problèmes posés :

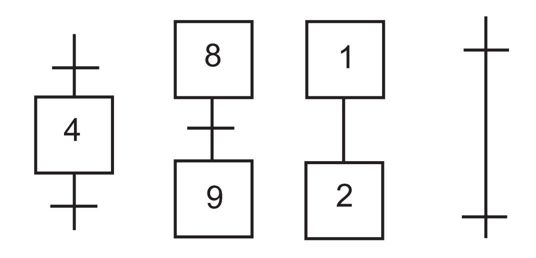
a) **Avec un tel flux d'informations peut-on utiliser la logique cablée pour gérer le pilotage automatique de ce système ?**
b) **Quelle logique peut-on utiliser ?**
c) **Comment utiliser et mettre en œuvre cette nouvelle technologie ?**

B- RAPPEL SUR LE GRAFCET

Activité
Réaliser l'activité N°1 du TP-A41 dans le manuel d'activités
Il s'agit de faire tourner un système automatique à fonctionnement cyclique, d'observer son fonctionnement et de le décrire sur un cycle complet.

1- Règles de syntaxe

L'alternance étape-transition et transition-étape doit toujours être respectée quelle que soit la séquence parcourue.
Deux étapes ou deux transitions ne doivent jamais être reliées par une liaison orientée. La liaison orientée relie obligatoirement une étape à une transition ou une transition à une étape.



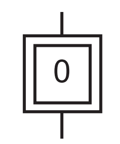
2- Règles d'évolution

a/ Règle N°1 : Situation initiale

La situation initiale d'un grafcet caractérise le comportement initial de la **PC** vis-à-vis de la **PO**, de l'opérateur et/ou des éléments extérieurs. Elle correspond aux étapes actives au début du fonctionnement. Elle traduit généralement un comportement de repos.
Une étape initiale est représentée par un carré double.

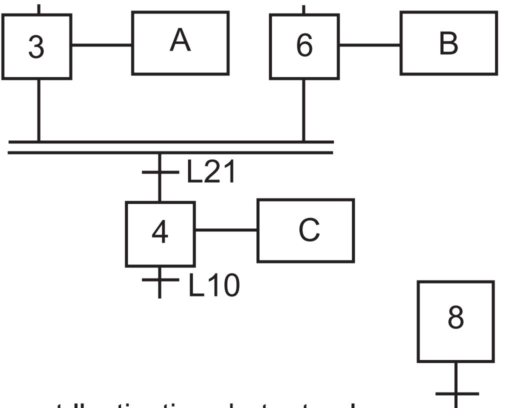
Notes :

- La situation initiale peut être atteinte :
  *soit lors de la mise en énergie de la **PC,**
  *soit naturellement en fin de cycle,
  *soit à la suite d'un forçage de situation.
- Un GRAFCET contient au moins une **ETAPE INITIALE**



b/ Règle N°2 : Évolution des étapes actives

Une transition est dite validée lorsque toutes les étapes immédiatement précédentes reliées à cette transition sont actives.
Le franchissement d'une transition se produit :
*lorsque la transition est validée,
*et que la réceptivité associée à cette transition est vraie.



c/ Règle N°3 : Franchissement d'une transition

Le franchissement d'une transition entraîne simultanément l'activation de toutes les étapes immédiatement suivantes et la désactivation de toutes les étapes immédiatement précédentes (évolution synchrone).

d/ Règle N°4 : Evolutions simultanées

Plusieurs transitions simultanément franchissables sont simultanément franchies.
Cette Règle 4 autorise la décomposition d'un GRAFCET unique en plusieurs GRAFCETS, tout en assurant de façon rigoureuse leur synchronisation. Pour cela, il faut intervenir dans les réceptivités des états actifs des étapes.

e/ Règle N°5 : Activation et désactivation simultanées d'une étape

Si, au cours du franchissement, la même étape est simultanément activée et désactivée elle reste active (priorité à l'activation).

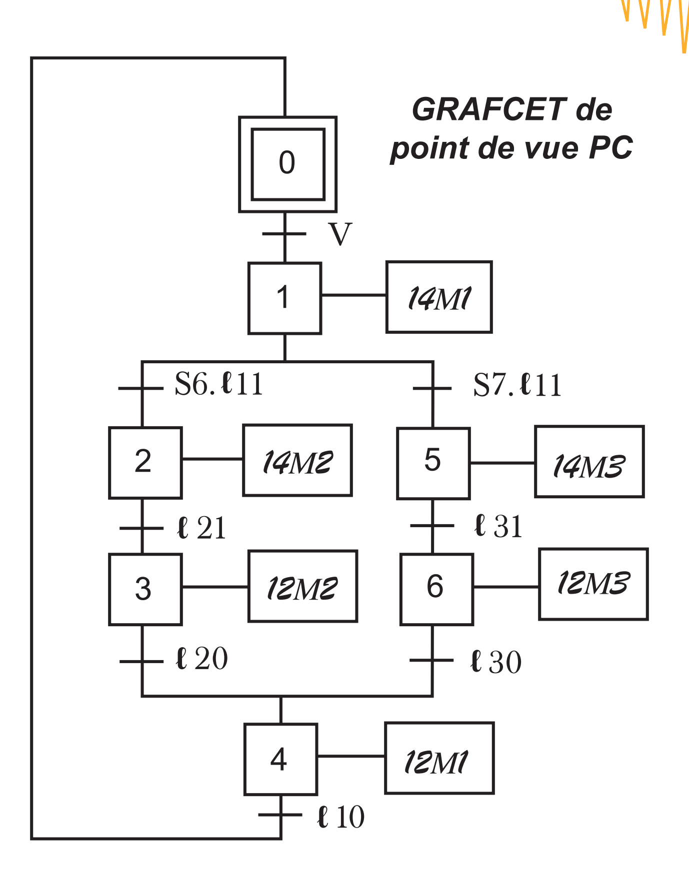
3- Exemple d'application : Système de triage des pièces

Un dispositif de triage doit permettre la répartition de deux types de pièces dans des casiers différents. Les pièces type 1 ou type 2 arrivent par gravité dans un ordre quelconque, mais l'encoche est toujours située à gauche de façon à être détectée par les capteurs S6 et S7.
On donne le GRAFCET de point de vue système et le tableau des affectations des entrées/sorties et on demande de tracer le Grafcet de point de vue PC.

pièces type 1
pièces type 2
C1
C2
V
S7
S6
C3
Casier pièces 2
Casier pièces 1

0
Présence pièce
1 — Déplacer la pièce
Pièce de type 1 déplacée | Pièce de type 2 déplacée
2 — Amener la pièce Dans le casier 1 | 3 — Amener la piéce Dans le casier 2
Pièce amenée dans le casier 1 | Pièce amenée dans le casier 2
4 — Initialiser le système
Système initialisé

Tableau des Affectations	Entrées système	Sorties système
Capteur Présence pièce	v	
Pièce type 1	s_6	- - - - -
Pièce type 2	s_7	- - - - -
Capteur sortie Vérin C1	ℓ_{11}	14M1
Capteur retour Vérin C1	ℓ_{10}	12M1
Capteur sortie Vérin C2	ℓ_{21}	14M2
Capteur retour Vérin C2	ℓ_{20}	12M2
Capteur sortie Vérin C3	ℓ_{31}	14M3
Capteur retour Vérin C3	ℓ_{30}	12M3



C- AUTOMATES PROGRAMMABLES INDUSTRIELS

Activité

Réaliser l'activité N°2 du TP-A41 dans le manuel d'activités

Il s'agit de faire tourner le même système avec un automate programmable, d'observer son fonctionnement et d'identifier les nouveaux éléments de l'installation.

1- Limites de la logique câblée :

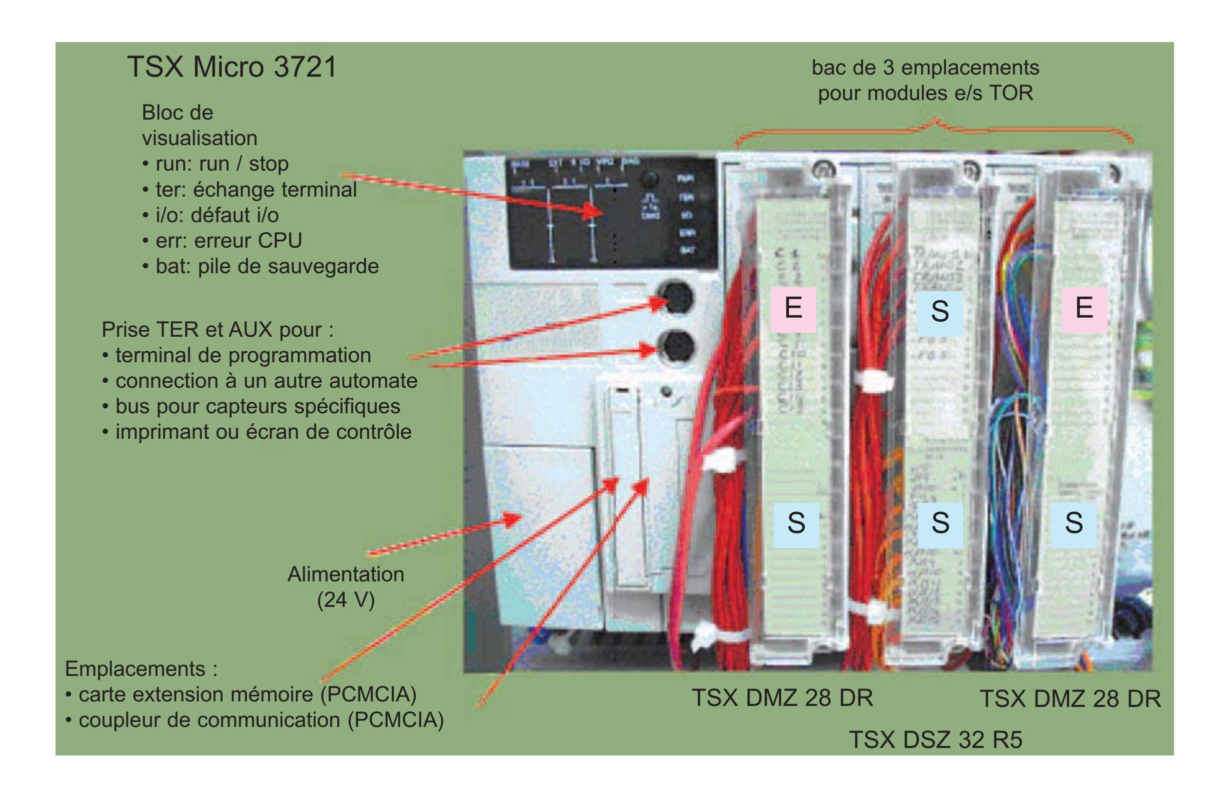
La synthèse directe d'un GRAFCET au moyen de composants cablés suppose implicitement que les composants mémoires soient :

- tout d'abord "synchronisés" (règles 3 et 4)
- ensuite prioritaires à l'activation (règle 5)

La réalisation en logique câblée, représente une solution figée qui nécessite d'être reprise intégralement en cas de modification du cycle de fonctionnement du système automatisé.
Les réalisations à partir de constituants modulaires industriels, dénommée "séquenceurs" ne sont pas toujours conformes à l'interprétation algébrique du GRAFCET.
Ces réalisations basées sur une interprétation dite " appel-réponse " ne respectent pratiquement aucune des règles formelles du GRAFCET (Les règles 3 à 5 ne sont jamais satisfaites).
Ces réalisations à base de séquenceurs sont **néanmoins** adaptées à des applications simples pour lesquelles il n'est pas demandé la rigueur du modèle formel.

2- Constitution d'un automate programmable :

a/ Exemple d'automate : le TSX 3721 de Télémécanique



b/ Constitution de base d'un Automate Programmable :

Un Automate Programmable Industriel est généralement construit autour de la structure de base suivante :

* un bloc des Entrées/Sorties TOR
* un bloc de commande et de visualisation
* une alimentation qui sert tous les autres blocs en énergie électrique
* des prises pour diverses connexions (terminal de programmation, autres automates, etc...)

3- Conception générale d'un API

" Un **automate programmable industriel** (en abrégé : API) est un **ensemble électronique** fonctionnant de manière **numérique** destiné à être utilisé pour **commander un processus industriel** (agro-alimentaire, fonderie, centre de tri, etc.). Il possède une **mémoire programmable** dans laquelle on range, à l'aide d'un **langage approprié**, toutes les **instructions** concernant le cycle du processus à automatiser. L'API et ses **périphériques** associés sont conçus pour pouvoir facilement s'intégrer à un système d'automatisme industriel et être facilement utilisés dans toutes leurs fonctions prévues. "

a/ Définition :

À partir de cette définition, on distingue les fonctions que l'API doit remplir :
– Un rôle de commande où il est un composant d'automatisme, élaborant les actions à la PO du processus à partir des informations que lui fournissent des capteurs ;
– Un rôle de communication
 * avec des opérateurs humains : c'est le dialogue d'exploitation,
 * avec d'autres processeurs (hiérarchiquement supérieurs pour la supervision par exemple) ou autres API intervenant dans la même chaîne de production.

b/ Forme commerciale :

Il existe de nombreuses marques d'automates. Parmi les plus courantes : Schneider electronic (ex. Téléméanique), Siemens, Allen-bradley, Modicon, Omron, Cégélec, Joucomatic, AEG, etc…

Exemples d'Automates Programmables :



Schneider Electronic (TSX)



Siemens



Allen Bradley



Omron



Modicon

c/ Raccordement de l'API

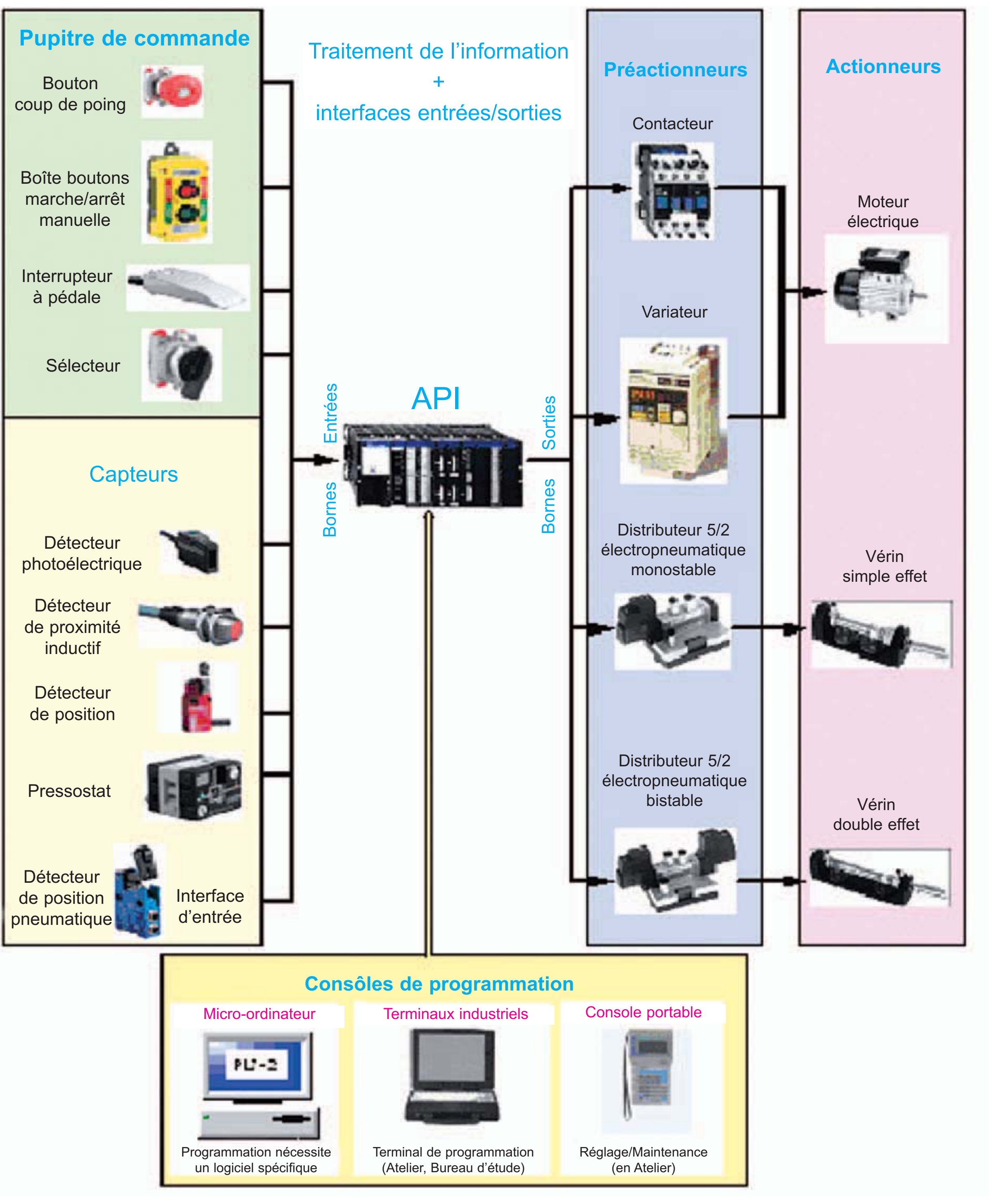
Activité

Réaliser l'activité N°3 du TP-A41 dans le manuel d'activités

Il s'agit de configurer correctement un automate programmable, en se référant aux documents constructeur et d'identifier les éléments du système raccordés à ses entrées/sorties.

– Les entrées de l'automate sont reliées avec les différents capteurs, détecteurs, accessoires de commande manuelles (boutons, commutateurs et claviers) et des sorties d'autres automates et circuits logiques qui interviennent dans la commande du processus industriel.

– Les sorties de l'automate sont reliées au moyen d'interfaces avec les différents actionneurs du processus industriel.



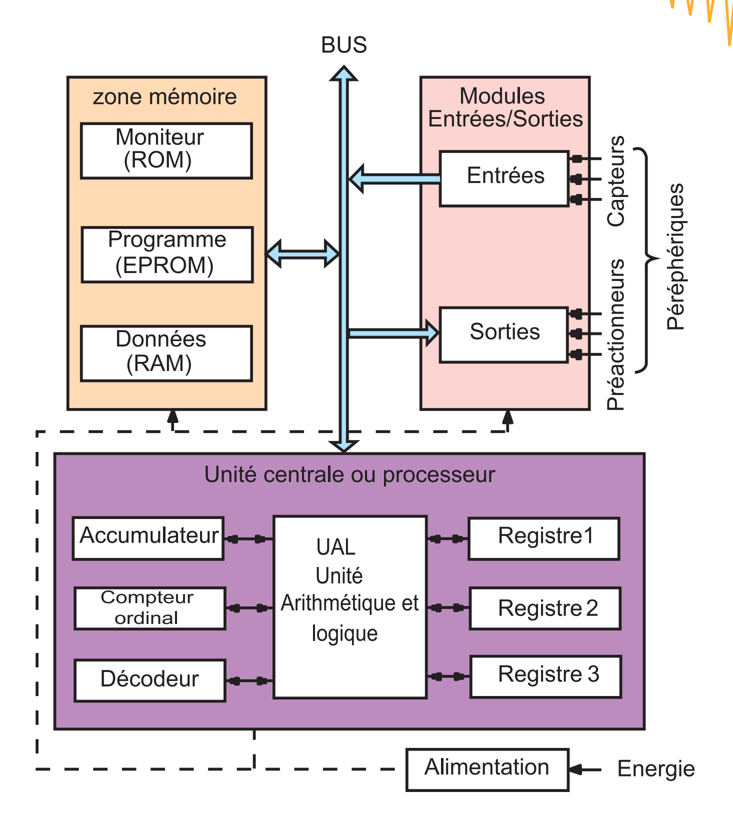
4- Architecture générale interne d'un API

Un API se compose, de trois grandes parties :

- un processeur (ou unité centrale) ;
- une zone mémoire ;
- des modules Entrées/Sorties.

La liaison électrique entre ces éléments est réalisée par un bus. Le bus est un ensemble de conducteurs qui réalisent la liaison entre les différents éléments de l'API. Il est organisé en plusieurs sous-ensembles destinés chacun à véhiculer un type bien défini d'informations :

– **Bus de données** pour les signaux d'entrées/sorties,
– **Bus d'adresses** des entrées/sorties,
–**Bus de contrôle** pour les signaux de service tels que tops de synchronisation, sens des échanges, contrôle de validité des échanges etc.
Un bloc d'alimentation fournit les tensions nécessaires au fonctionnement de l'ensemble.



a/ Le Processeur

Le processeur, ou unité centrale (UC), a pour rôle principal d'interpréter et d'exécuter les instructions qui constituent le programme de fonctionnement de l'application. Il gère l'ensemble des relations entre la zone mémoire et les interfaces telles que :

- lecture des informations d'entrée ;
- exécution des instructions du programme mises en mémoire ;
- commande des sorties.

Pour réaliser ces fonctions, le processeur doit comporter :

- Une unité arithmétique et logique (UAL) qui traite des opérations logiques (ET, OU,…) et des opérations arithmétiques (addition, soustraction, décalage, etc...) ;
- Un accumulateur, registre de travail dans lequel se range le résultat d'une opération ;
- Un compteur ordinal : registre qui contient l'adresse de la prochaine instruction à exécuter ;
- Un décodeur d'instruction : circuit qui transforme les instructions d'un programme en signaux de commande nécessaires à l'UAL, aux registres et au bus de contrôle;
- Des registres dans lesquels sont rangées durant le temps de traitement les instructions à interpréter et à exécuter.

Il existe trois technologies pour réaliser un processeur :

- La technologie câblée, la plus rapide mais aussi la plus coûteuse, réservée à des usages particuliers ;
- La technologie à microprocesseur, la plus économique ;

✔ La technologie mixte, certaines opérations étant réalisées en câblé pour en accroître la rapidité.

b/ Zone mémoire

La zone mémoire est destinée au stockage des données (informations issues des capteurs d'entrée et informations générées par le processeur pour commander les sorties) et des programmes de fonctionnement de l'automatisme.
L'élément de base de la mémoire est le " BIT " (abréviation de l'anglais " Binary digiT") qui peut prendre les deux états logiques 0 et 1. Les bits sont regroupés en "mots ". On rencontre des mots de 16 bits et de 8 bits appelés " octets ". Ces mots sont identifiés chacun par une adresse. Le volume mémoire est évalué en " K mots " (1 K mot = 2^{10} mots = 1 024 mots) ou en " Koctets " (1 Ko = 1 024 mots de 8 bits).

Dans les API, on trouve plusieurs types de mémoires :

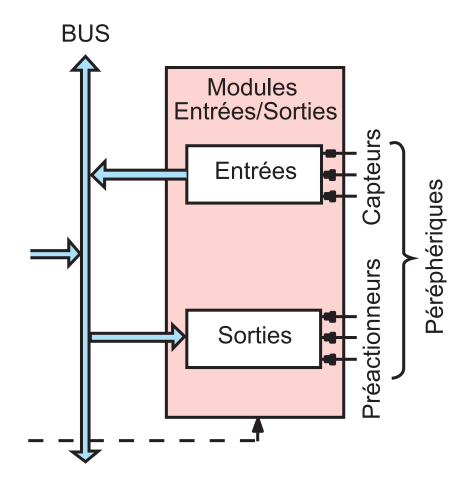
* **Mémoires vives ou RAM** (Random Access Memory) : le contenu de ces mémoires peut être lu et modifié à volonté, mais il est perdu en cas de manque de tension (mémoires volatiles). Elles nécessitent par conséquent une sauvegarde par batterie. Les mémoires vives sont utilisées pour l'écriture et la mise au point du programme et le stockage des calculs ; des variables; etc ...

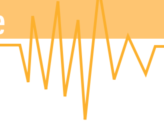
* **Mémoires non volatiles effaçables électriquement** (EEPROM : Electrically Erasable Programmable Read Only Memory) ou par rayonnement ultraviolet (EPROM ou REPROM : Erasable Programmable Read Only Memory). Elles sont destinées à la mémorisation du programme après la phase de mise au point.

* **Mémoires mortes ou ROM** (Read Only Memory) : Dans ces mémoires seule la lecture est possible. Elles servent, pour le constructeur de l'API, à mémoriser le programme de fonctionnement de l'Automate Programmable Industriel (l'API).

c/ Modules d'entrées/sorties (E/S)

On considère ici que les modules d'entrées/sorties sont du type" **Tout Ou Rien " (TOR**). Ces modules assurent le rôle d'interface des capteurs et préactionneurs avec l'API.
Les entrées reçoivent des informations en provenance des éléments de détection (capteurs) et du pupitre opérateur (boutons). Chaque capteur est repéré par une adresse d'entrée (par exemple pour le TSX : I1.0, I1.1, I1.2…). L'ensemble des modules d'entrées constitue l'interface d'entrée.
Les informations de sortie sont dirigées vers les modules constituant l'interface de sortie. Chaque module est repéré par une adresse de sortie (par exemple pour le TSX : Q2.0, Q2.1, Q2.2…). Les éléments de commande de puissance (électrovanne, relais…), les éléments de contrôle du pupitre opérateur (voyants, compteur, alarme) sont reliés à ces sorties.





Modules d'entrée TOR

Les modules d'entrée sont destinés à :

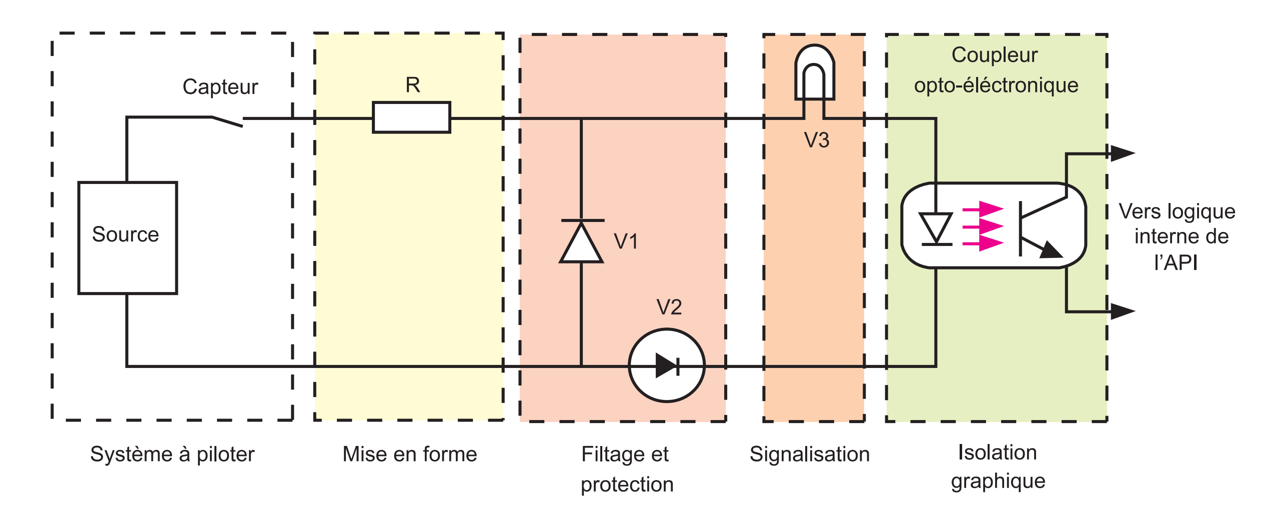
* Recevoir l'information en provenance de la source d'information (capteur, détecteur, bouton etc...). Le signal électrique reçu doit être analysé par le module d'entrée avant d'être transmis à l'unité centrale par le Bus.
* Traiter le signal en le mettant en forme, en éliminant les parasites et en isolant électriquement les signaux de la partie opérative par un coupleur opto-électronique.

Dans ce but, un module d'entrée effectue les fonctions suivantes :

* Adaptation de la tension ;
* Immunité aux parasites industriels par filtrage et circuit retardateur (temporisateur) ;
* Signalisation de l'état de l'entrée;
* Isolement entre l'extérieur et la partie logique de l'API.

Schéma de principe d'une entrée TOR

avec : **R** : adaptation au seuil de tension de l'entrée.
V1 : protection en cas d'inversion de tension aux entrées.
V2 : retardateur, diode zéner en écrêteur.
V3 : signalisation de l'état de l'entrée.
Coupleur opto-électronique qui est l'association d'une diode électroluminescente et d'un photo-transistor.



Modules de sortie tout ou rien (TOR)

Les modules de sortie sont destinés à convertir les signaux de " bas niveau " de la partie logique en signaux utilisables pour piloter les préactionneurs (électrovannes, contacteurs).

Les caractéristiques des sorties dépendent de la technologie utilisée :

* Électromécanique (relais) :
  - Fort pouvoir de coupure,
  - Cadence de fonctionnement limitée,
  - Durée de vie limitée,
  - Découplage galvanique possible ;
* Électronique (transistor, triac) :
  - Cadence de fonctionnement élevée,
  - Durée de vie pratiquement illimitée.

Un module de sortie assure les fonctions suivantes :

– l'isolation galvanique
– l'adaptation en tension
– la protection contre le retour de courant
– et la visualisation de l'état logique de la sortie

L'élément de commande en sortie sera constitué selon le cas d'un relais, d'un transistor ou d'un triac.

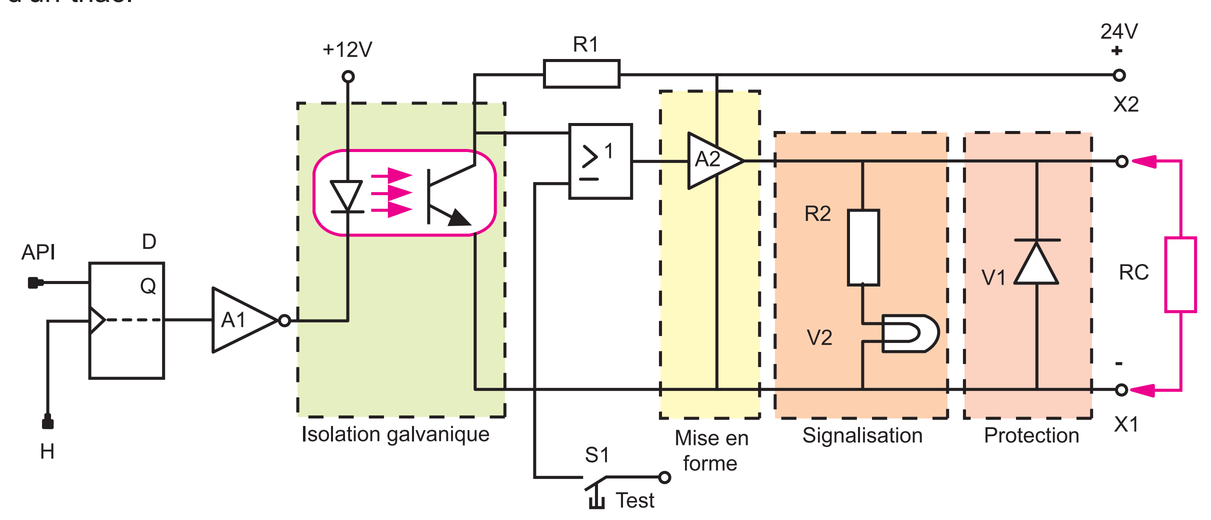


Schéma de principe d'une sortie logique 24 V continue

D : mémorisation du résultat du traitement.
A1, A2 : amplification.
V1 : protection en cas d'inversion de tension aux bornes de la sortie.
V2 : signalisation de l'état de la sortie.
RC : charge.
X1, X2 : bornes de la source d'alimentation extérieure.

Suivant la norme NF C 63850, les sorties logiques doivent satisfaire les caractéristiques suivantes :

– en courant alternatif 50 Hz, sous des tensions de 24, 48, 110 et 220 V avec un courant maximal de 0,25 A, 1 A, 2 et 5 A ;

– en courant continu, sous des tensions de 5, 24, 48, 110 et 125 V avec un courant permanent maximal de 0,25 A, 1 et 2 A.



5- Programmation d'un API

Activité
Réaliser l'activité pratique N°4 ou N°4Bis du TP-A41 dans le manuel d'activités
Il s'agit de traduire un GRAFCET codé automate en un programme liste d'instructions (LI) permettant de piloter un système de commande d'un moteur asynchrone dans les deux sens de marche.

a/ Programme

Un programme est l'unité logique de programmation qui décrit des opérations entre les variables de l'application. Un programme décrit des opérations séquentielles ou cycliques. Un programme cyclique est exécuté systématiquement à chaque cycle automate. L'exécution d'un programme séquentiel respecte les règles d'évolution du GRAFCET.

b/ Déroulement d'un programme

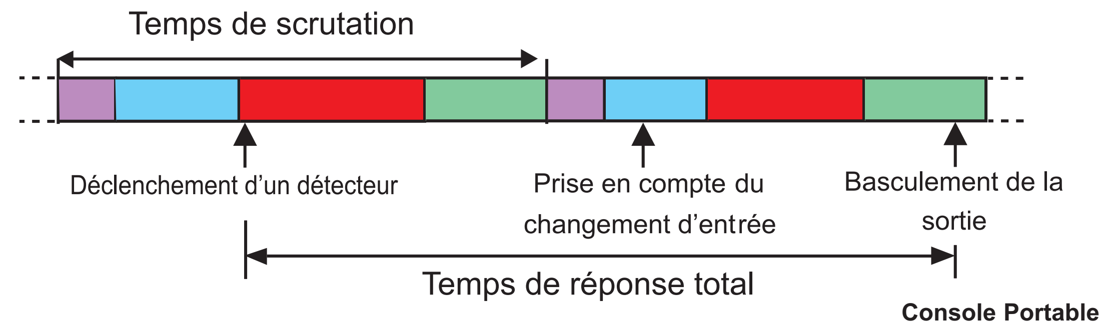
Un programme doit assurer en permanence un cycle opératoire qui comporte trois types de tâches :

* L'acquisition de la valeur des entrées (lecture) ;
* Le traitement des données ;
* L'affectation de la valeur des sorties (écritures).

Le premier et le troisième point s'effectuent par mise en communication successive de chacune des voies d'entrée ou de sortie (scrutation).

Le temps de scrutation, qui sépare deux lancements successifs de la même portion du programme d'application, comprend non seulement le temps de traitement mais aussi le temps de mise à jour de la mémoire des E/S.

Le temps de réponse total (TRT) est le temps qui s'écoule entre le changement d'état d'une entrée et le changement d'état de la sortie correspondante.



c/ Moyens de programmation

Leur rôle consiste à :

* Écrire et modifier le programme de l'API ;
* Le stocker ;
* Le visualiser en temps réel via l'état des E/S et variables intermédiaires des algorithmes de commande ;
* Modifier certaines données ;
* Aider au diagnostic des défauts

Console Portable



Réglage/Maintenance (en Atelier)

Ils se composent d'une partie matérielle (console ou ordinateur) et d'un logiciel de programmation.

- Consoles de programmation : terminaux de poche, ces outils permettent, par connexion directe sur site, de modifier des valeurs de temporisation de compteurs, des bits ou des mots, parfois le programme lui-même. Elles sont dédiées à une marque d'appareils par exemple :
T407, T607, FTX2000 Télémécanique
PG610, 630, 710, 730, 750, etc de la gamme Siemens

- Micros ordinateurs équipés du logiciel de programmation propre à la gamme d'un constructeur.
Exemples : XTEL, PL7 Micro, PL7 Pro Télémécanique STEP7 pour les API SIEMENS



- Atelier logiciel indépendant de la marque au stade de l'élaboration fonctionnelle, suivi d'un " post-processeur " spécifique à l'automate.
Exemples : CADEPA, ISAGRAF



Terminaux Industriels

d/ Langages de programmation

La norme internationale CEI 1131-3 reprise dans la norme française EN 61131-3 définit cinq types de langage du monde industriel, à savoir :

** IL (Instruction List) ou liste d'instructions : ce langage textuel de bas niveau est un langage à une instruction par ligne. Il ressemble, dans certains aspects, au langage assembleur employé pour la programmation des microprocesseurs ;

** ST (Structured Text) ou texte structuré : ce langage est un langage textuel de haut niveau. Il permet la programmation de tout type d'algorithme plus ou moins complexe;

** LD (Ladder Diagram) ou schéma à contacts : ce langage graphique est essentiellement dédié à la programmation d'équations booléennes (true/false) ;

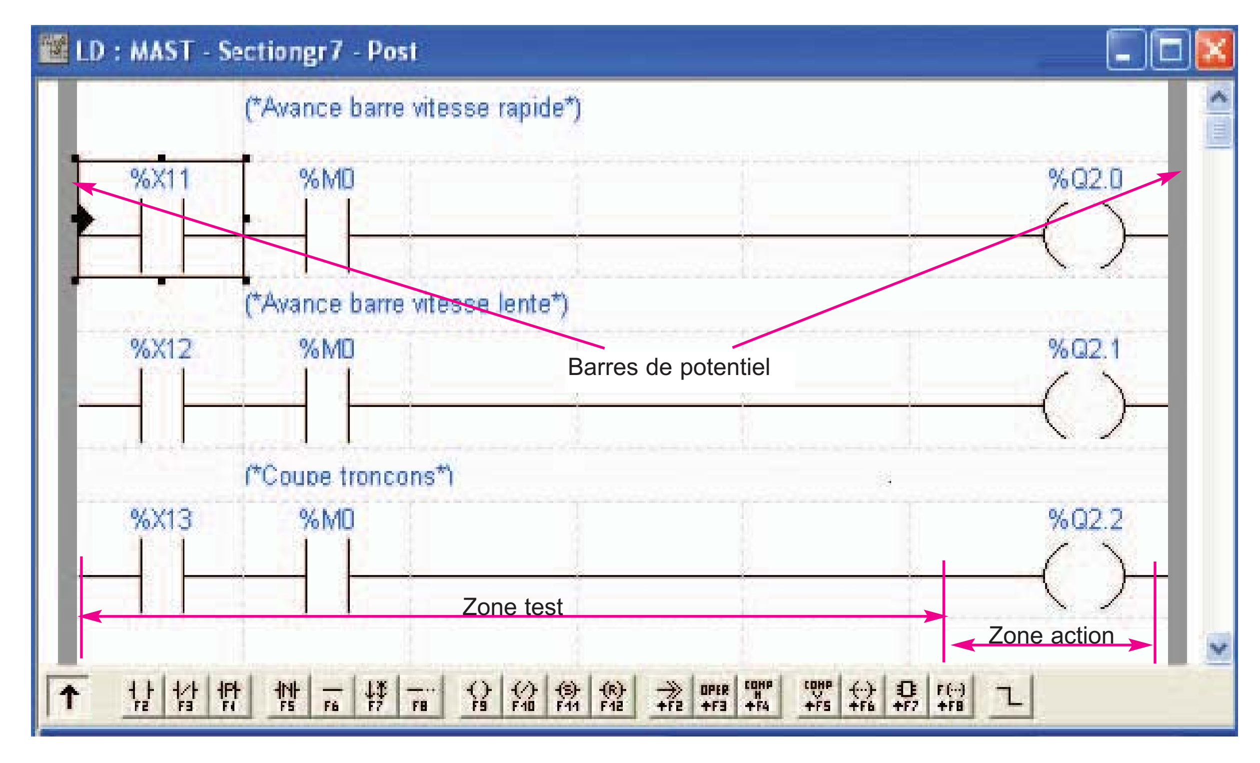
** SFC (Sequential function Chart) : issu du langage GRAFCET, ce langage de haut niveau permet la programmation aisée de tous les procédés séquentiels ;

** FBD (Function Block Diagram) ou schéma par bloc : ce langage permet de programmer graphiquement à l'aide de blocs, représentant des variables, des opérateurs ou des fonctions. IL permet de manipuler tous les types de variables.

- Programmation en réseaux de contacts (Ladder)

Une section de programme écrite en langage à contacts se compose d'une suite de réseaux de contacts exécutés séquentiellement par l'automate.La représentation d'un réseau de contacts est proche de celle d'un schéma électrique.

Description d'un réseau de contacts



Un réseau de contacts est composé d'un ensemble d'éléments graphiques disposés sur une grille organisée en lignes et colonnes.

Exemple : pour l'Automate TSX3721, on a :

– 16 lignes maximum et 11 colonnes (pour automates Premium),
– 7 lignes maximum et 11 colonnes (pour automates Micro).

Il est réparti en deux zones :
– la zone test, dans laquelle figurent les conditions nécessaires à une action
– la zone action, qui applique le résultat consécutif à un enchaînement de test.

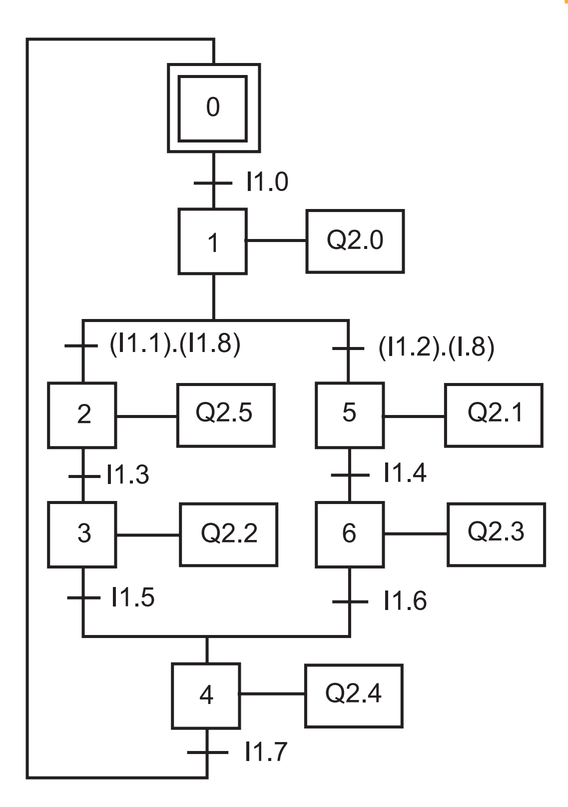
Exemple d'application :

En se référant au dossier technique du TSX 3721, reprendre l'exemple traité en A et dont on a tracé le GRAFCET PC et écrire un programme Ladder qui permet de commander le système de tri de pièces.

Solution :

Il faut d'abord commencer par transformer le GRAFCET PC en GRAFCET codé Automate en respectant le tableau des affectations ci-contre :

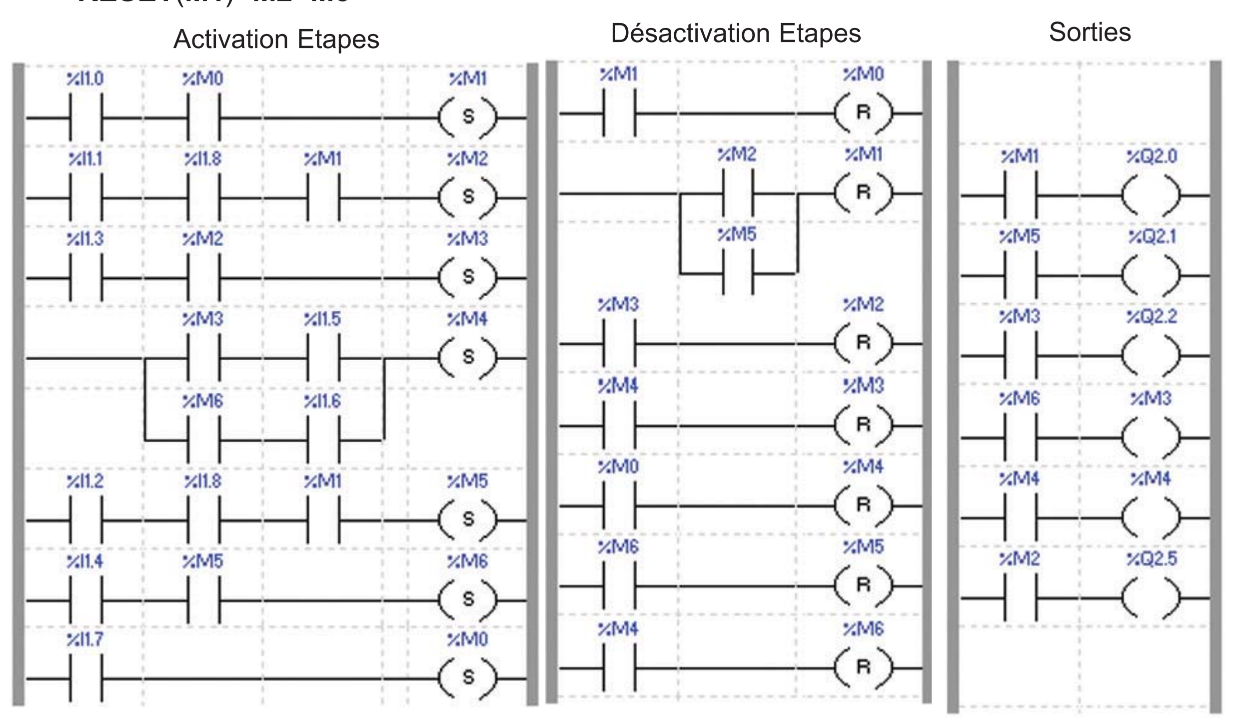
Tableau des Affectations	Entrées système	Entrées TSX	Sorties système	Sorties TSX
Capteur Présence pièce	u	I1.0		
Pièce type 1	s_6	I1.1	- - - - - -	
Pièce type 2	s_7	I1.2	- - - - - -	
Capteur sortie Vérin C1	ℓ_{11}	I1.8	14M1	Q2.0
Capteur retour Vérin C1	ℓ_{10}	I1.7	12M1	Q2.4
Capteur sortie Vérin C2	ℓ_{21}	I1.3	14M2	Q2.5
Capteur retour Vérin C2	ℓ_{20}	I15	12M2	Q2.2
Capteur sortie Vérin C3	ℓ_{31}	I1.4	14M3	Q2.1
Capteur retour Vérin C3	ℓ_{30}	I1.6	12M3	Q2.3



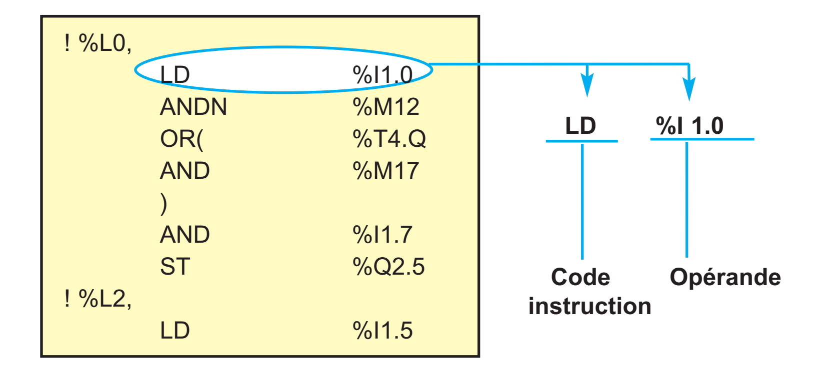
Les étapes sont programmées commes des bobines avec une mise à 1 et une remise à 0 :

* la mise à 1 (SET) est déduite de l'équation d'activation de l'étape.
* la mise à 0 (RESET) est déduite de l'équation de désactivation.

Ex : SET(M1)=M0.(I1.0)
RESET(M1)=M2+M5



Programmation en lignes d'instructions



Une section écrite en langage liste d'instructions se compose d'une suite d'instructions exécutées séquentiellement par l'automate.
L'illustration suivante présente un programme liste d'instructions PL7 et le détail d'une instruction.

Pour traduire un GRAFCET, codé Automate, en lignes d'instructions on passe par plusieurs étapes identiques à celles suivies précédemment dans l'élaboration des réseaux à contacts :

– Les étapes sont programmées commes des bobines avec une mise à 1 et une remise à 0.
  * la mise à 1 (**S**) est déduite de l'équation d'activation de l'étape.
  * la remise à 0 (**R**) est déduite de l'équation de désactivation.

Ex : avec le **TSX 3721**, la programmation de la mémoire étape1 est la suivante :

```
LD     %M0
AND    %I1 .0
S      %M1
LD     %M2
R      %M1
```

– Les sorties sont prises par équivalence avec les étapes.

Ex : avec le **TSX 3721**, la programmation de l'action associée à l'étape1 est la suivante :

```
LD     %M1
ST     %Q2.1
```

Exemple d'application :

En se référant au dossier technique du TSX 3721, reprendre l'exemple traité sur le système de tri de pièces et écrire un programme liste d'instructions qui permet de gérer sa commande.



Solution :

On garde les affectations des entrées/sorties choisies dans l'exemple, ainsi que le fonctionnement. Le programme tel qu'il est saisi sur le logiciel PL7 est le suivant :

Etape 0	LD	%M4
	AND	%I1.7
	S	%M0
	LD	%M1
	R	%M0

Etape 1	LD	%M0
	AND	%I1.0
	S	%M1
	LD	%M2
	R	%M1

Etape 2	LD	%M1
	AND	%I1.1
	AND	%I1.8
	S	%M2
	LD	%M3
	R	%M2

Etape 3	LD	%M2
	AND	%I1.3
	S	%M3
	LD	%M4
	R	%M3

Etape 4	LD	%M3
	AND	%I1.5
	OR(	%M6
	AND	%I1.6
	)	
	S	%M4
	LD	%M0
	R	%M4

Etape 5	LD	%M1
	AND	%I1.2
	AND	%I1.8
	S	%M5
	LD	%M6
	R	%M5

Etape 6	LD	%M5
	AND	%I1.4
	S	%M6
	LD	%M4
	R	%M6

14M1	LD	%M1
	ST	%Q2.0

14M2	LD	%M2
	ST	%Q2.5

12M2	LD	%M3
	ST	%Q2.2

12M1	LD	%M4
	ST	%Q2.4

14M3	LD	%M5
	ST	%Q2.1

12M3	LD	%M6
	ST	%Q2.3



✔ Programmation en langage GRAFCET

Activité

Réaliser l'activité N°5 du TP-A41 dans le manuel d'activités

Il s'agit de traduire un GRAFCET codé automate en un programme langage GRAFCET permettant de piloter un système de commande d'un moteur asynchrone dans les deux sens de marche.

Le langage GRAFCET n'est pas utilisé avec tous les types d'Automates Programmables disponibles dans l'industrie. Il est développé dans ce chapitre à cause des exercices traités autour du TSX 3721 de Télémécanique.

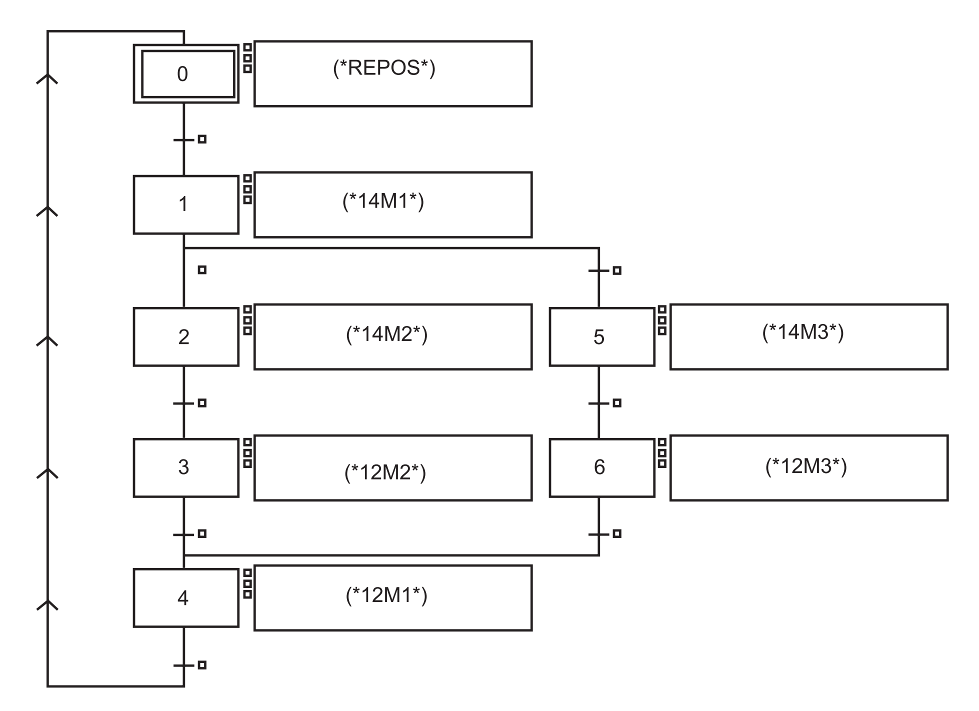
Le logiciel PL7 dispose :

– d'objets bits associés aux étapes (1=étape active)
  %Xi Etat de l'étape i du Grafcet principal(i de 0 à n, n dépend du processeur)
– de bits système spécifiques
  %S21 Provoque l'initialisation du Grafcet
  %S22 Provoque la remise à zéro générale du Grafcet
  %S23 Provoque le figeage du Grafcet

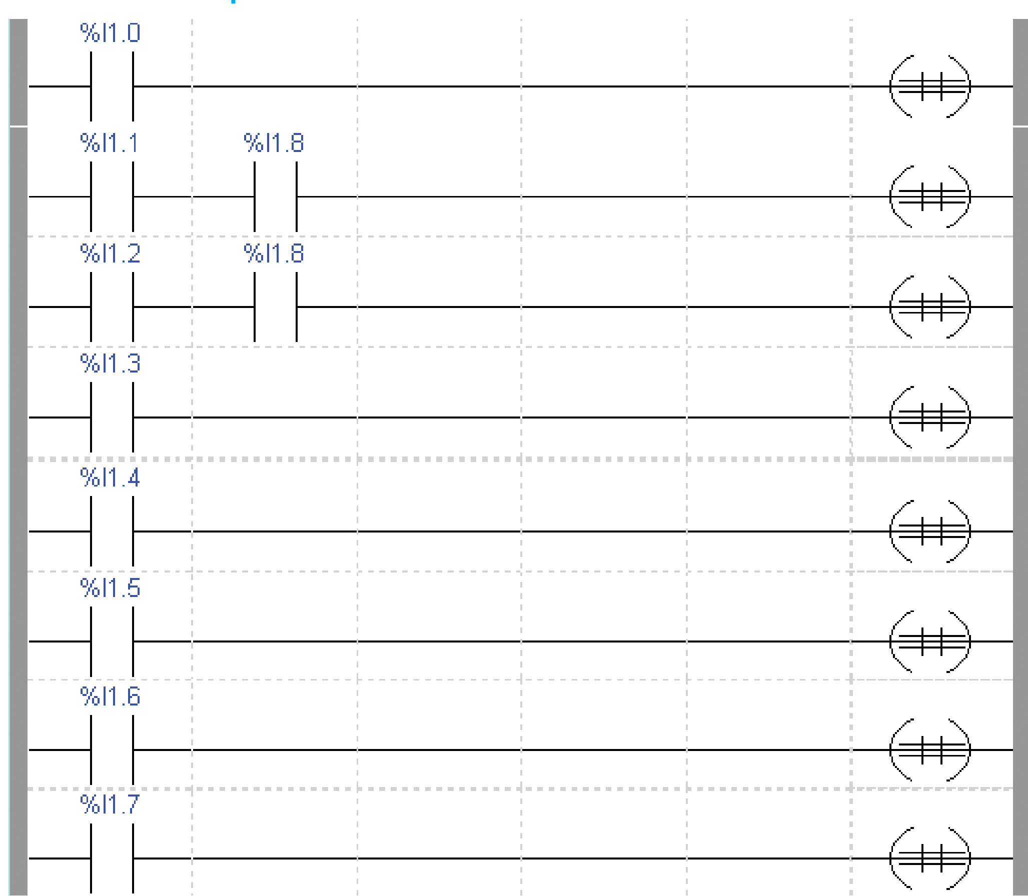
Exemple d'application :

En se référant au dossier technique du TSX 3721, reprendre l'exemple traité avec le langage liste d'instructions et écrire un programme GRAFCET qui permet de commander le système de tri de pièces.

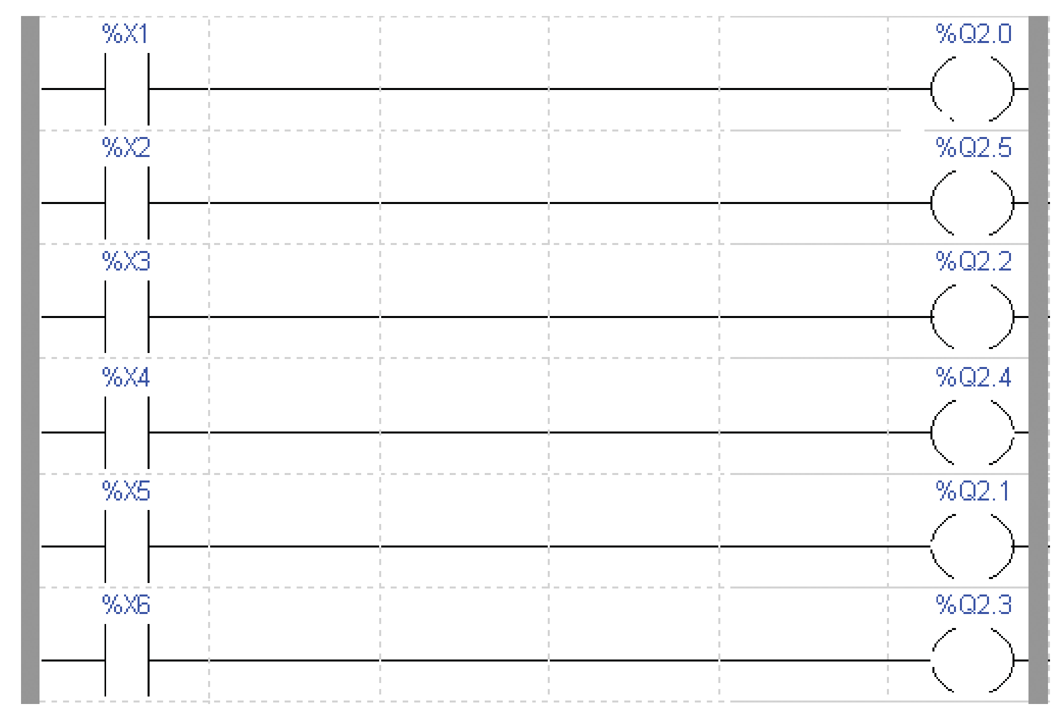
Solution : Construction du GRAFCET



Construction des receptivités



Programmation des actions associées



D- RESUMÉ :

Un **GRAFCET** est établi à partir d'un **cahier des charges**. Il doit respecter les **règles** d'écriture :

- une règle de **syntaxe**
- cinq règles **d'évolution**

Pour des **processus complexes**, la **logique cablée** devient **insuffisante**. Elle est remplacée par la **logique programmée**.

Un **automate programmable industriel** est un **ensemble électronique** fonctionnant de manière **numérique** destiné à être utilisé pour **commander un processus industriel**. Il possède une **mémoire programmable** dans laquelle on range, à l'aide d'un **langage approprié**, toutes les **instructions** concernant le cycle du processus à automatiser. L'API et ses **périphériques** associés sont conçus pour pouvoir facilement s'intégrer à un système d'automatisme industriel et être facilement utilisés dans toutes leurs fonctions prévues.

A partir du **GRAFCET PC**, et en **fonction de l'Automate Programmable** choisi pour le pilotage du système, on peut écrire un **GRAFCET codé automate**.

Pour traduire le **GRAFCET** codé automate en **programme**, plusieurs **langages** sont utilisés dont essentiellement :

- langage **liste d'instructions**
- langage **réseau à contact**
- langage **GRAFCET**
- langage **texte structuré**
- **etc...**

Dans **le bus** équipé d'un **copilote électronique**, les capteurs transmettent les informations à **un automate programmable** installé dans un coffret localisé derrière le conducteur. Après **traitement,** l'automate **envoie** ces **informations** d'une part à un modem radio pour **transmission** au dépôt et d'autre part à un **écran de dialogue.** Cet écran, positionné dans le champ visuel du conducteur, fournit toute l'aide nécessaire à la manœuvre d'arrêt. Il comporte une **rampe lumineuse horizontale,** constituée par **8 voyants lumineux,** qui permet au conducteur de connaître sa **position**, **trop à gauche** ou **trop à droite** de la ligne blanche.

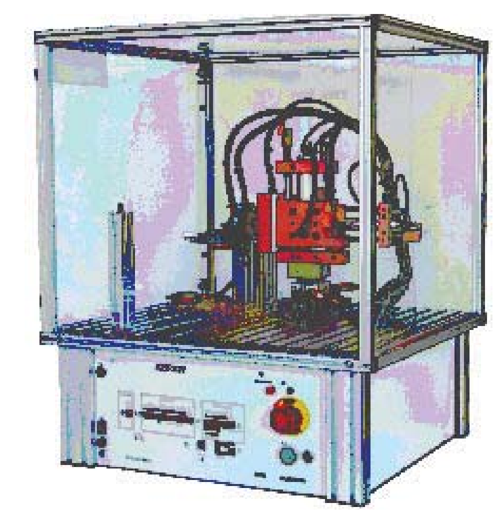
E- EXERCICES D'APPLICATION :

Exercice à résoudre :

Présentation du système : Unité de transfert
Le robot électropneumagnétique **NEWMAT** est un bras manipulateur 5 axes.
On désire concevoir les circuits de commande du Robot de façon qu'il assure le transfert d'une pièce du stock au poste 1 vers la zone de marquage au poste 2 et qu'il la ramène au même endroit là où il l'a prise conformément à la chronologie suivante :

- le bras avance avec pince ouverte
- la pince se ferme pour prendre la pièce
- la base tourne à droite

Après dépose de la pièce par ouverture de la pince :
- le bras recule en gardant la pince ouverte
- le bras avance
- la pince se referme
- la base tourne à gauche
- la pince s'ouvre de nouveau pour déposer la pièce
- le bras recule vers la position initiale



Robot Newmat

1) Etablir le GRAFCET de point de vue de la partie commande en se référant au tableau des choix technologiques suivant :

Liste des informations		Liste des actions	
Label	**Désignation**	**Label**	**Désignation**
dcy	Départ cycle	12M1	Ouverture pince
ℓ_{20}	Base robot à gauche	14M1	Fermeture pince
ℓ_{21}	Base robot à droite	14M2	Rotation Droite
ℓ_{10}	Pince ouverte	12M2	Rotation Gauche
ℓ_{11}	Pince fermée	14M3	Avance Bras
ℓ_{31}	Bras en avant	12M3	Recul Bras
ℓ_{30}	Bras en arrière		

2) On désire implanter ce GRAFCET sur l'Automate Programmable industriel TSX3721.
Etablir le GRAFCET codé automate en respectant le tableau des affectations présenté ci-contre.

3) Traduire le GRAFCET codé automate en un programme écrit en liste d'instructions relatives à l'automate utilisé.

Tableau des Affectations	Entrées système	Entrées TSX	Sorties système	Sorties TSX
Bouton Dépat Cycle	dcy	I1.0		
Fermeture Pince	ℓ_{11}	I1.8	14M1	Q2.0
Ouverture Pince	ℓ_{10}	I1.7	12M1	Q2.4
Base Robot à Droite	ℓ_{21}	I1.3	14M2	Q2.5
Base Robot à Gauche	ℓ_{20}	I15	12M2	Q2.2
Bras en Avant	ℓ_{31}	I1.4	14M3	Q2.1
Bras en Arrière	ℓ_{30}	I1.6	12M3	Q2.3

• Eléments graphiques du langage à contacts

Généralités

Les éléments graphiques sont les instructions du langage à contacts.

Contacts

Les éléments graphiques des contacts se programment en zone test et occupent une cellule (1 ligne de hauteur et 1 colonne de largeur).

<table>
<tr><th>Désignation</th><th>Graphisme</th><th>Fonctions</th></tr>
<tr><td>Contact à fermeture</td><td>—| |—</td><td>Contact passant quand l'objet bit qui le pilote est à l'état 1.</td></tr>
<tr><td>Contact à ouverture</td><td>—|/|—</td><td>Contact passant quand l'objet bit qui le pilote est à l'état 0.</td></tr>
<tr><td>Contact à détection de front montant</td><td>—|P|—</td><td>Front montant : détection du passage de 0 à 1 de l'objet bit qui le pilote.</td></tr>
<tr><td>Contact à détection de front descendant</td><td>—|N|—</td><td>Front descendant : détection du passage de 1 à 0 de l'objet bit qui le pilote.</td></tr>
</table>

Eléments de liaison

Les éléments graphiques de liaison permettent de relier les éléments graphiques de test et d'action.

<table>
<tr><th>Désignation</th><th>Graphisme</th><th>Fonctions</th></tr>
<tr><td>Connexion horizontale</td><td>——</td><td>Permet de relier en série les éléments graphiques de test et d'action entre les deux barres de potentiel.</td></tr>
<tr><td>Connexion verticale de potentiel</td><td>|</td><td>Permet de relier en parallèle les éléments graphiques de test et d'action.</td></tr>
<tr><td>Dérivation court-circuit</td><td>‾‾|__</td><td>Permet de relier 2 objets au travers de plusieurs connexions.</td></tr>
</table>

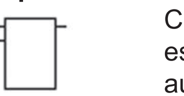
Bobines

Les éléments graphiques des bobines se programment en zone action et occupent une cellule (1 ligne de hauteur et une colonne de largeur).

Désignation	Graphisme	Fonctions
Bobine directe	-( )-	L'objet bit associé prend la valeur du résultat de la zone test.
Bobine inverse	-(/)-	L'objet bit associé prend la valeur inverse du résultat de la zone test.
Bobine d'enclenchement	-(S)-	L'objet bit associé est mis à 1 lorsque le résultat de la zone test est à 1.
Bobine de déclenchement	-(R)-	L'objet bit associé est mis à 0 lorsque le résultat de la zone test est à 1.
Saut conditionnel à un autre réseau (JUMP)	->>%Li	Permet un branchement à un réseau étiqueté, amont ou aval. Les sauts ne sont effectifs qu'au sein d'une même entité de programmation (programme principal, sous-programme,...). L'exécution d'un saut provoque : ✓ l'arrêt de la scrutation du réseau en cours, ✓ l'exécution du réseau étiqueté demandé, ✓ la non scrutation d'une partie du prog située entre l'action de saut et le réseau désigné.
Bobine dièse	-(#)-	Proposée en langage Grafcet, utilisée lors de dièse la programmation des réceptivités associées aux transistions provoque le passage à l'étape suivante.
Bobine appel à un sous-programme (CALL)	-(C)-	Permet un branchement en début de sous-programme lorsque le résultat de la zone de test sous-programme est à 1. L'exécution d'un appel à un sous-programme provoque : ✓ l'arrêt de la scrutation du réseau en cours, ✓ l'exécution du sous-programme, ✓ la reprise de la scrutation du réseau interrompu.
Retour de sous-programme	<RETURN>	Réservée au sous-programme SR, permet le retour au module appelant lorsque le résultat de la zone de test est à 1.
Arrêt programme	<HALT>	Provoque l'arrêt de l'exécution du programme lorsque le résultat de la zone de test est à 1.

Blocs fonction standard

Les éléments graphiques des blocs fonction standard se programment en zone test et occupent une dimension d'une hauteur de 16 lignes maximum et une largeur 3 colonnes.

Désignation	Graphisme	Fonctions
Blocs Temporisateur, Compteur, Monostable, Registre, Programmateur cyclique	 	Chacun des blocs fonctions standards utilise des entrées, es sorties, des entrées/sorties permettant de les reliés aux autres éléments graphiques

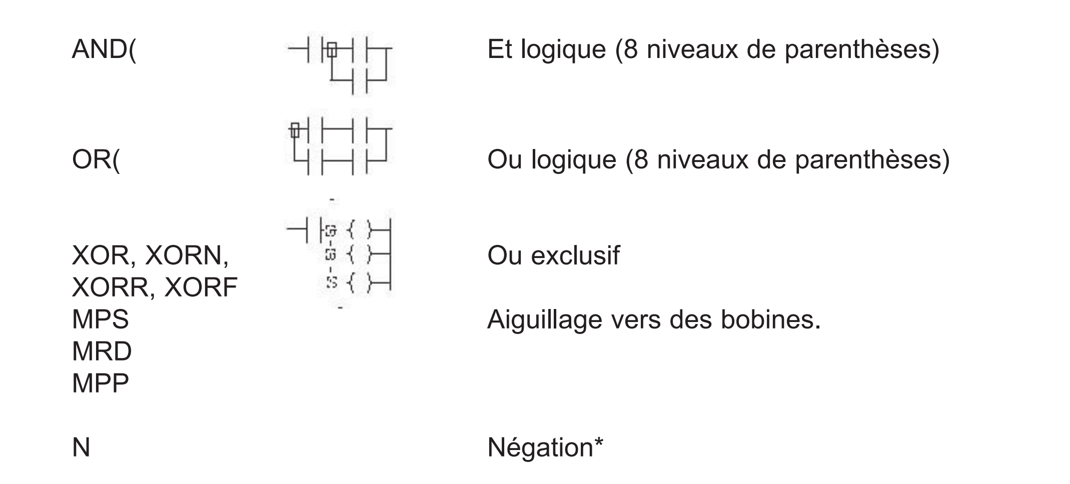
• Eléments graphiques du langage à contacts

Instructions de test

Le tableau suivant décrit les instructions de test du langage liste d'instructions..

Désignation	Graphisme équivalent	Fonctions
LD	—| |—	Le résultat booléen est égal à l'état de l'opérande.
LDN	—|/|—	Le résultat booléen est égal à l'état inverse de l'opérande.
LDR	—|P|—	Le résultat booléen passe à 1 à la détection du passage de 0 à 1 de l'opérande (front montant).
LDF	—|N|—	Le résultat booléen passe à 1 à la détection du passage de 1 à 0 de l'opérande (front descendant).
AND	—| |——| |—	Le résultat booléen = Et logique entre le résultat booléen de l'instruction précédente et l'état de l'opérande.
ANDN	—| |——|/|—	Le résultat booléen est égal au Et logique entre le résultat booléen de l'instruction précédente et l'état inverse de l'opérande.
ANDR	—| |——|P|—	Le résultat booléen est égal au Et logique entre le résultat booléen de l'instruction précédente, et la détection d'un front montant de l'opérande (1=front montant).
ANDF	—| |——|N|—	Le résultat booléen est égal au Et logique entre le résultat booléen de l'instruction précédente, et la détection d'un front descendant de l'opérande (1=front descendant).
OR	—| |— en parallèle avec —| |—	Le résultat booléen = Ou logique entre le résultat booléen de l'instruction précédente et l'état de l'opérande.
ORN	—| |— en parallèle avec —|/|—	Le résultat booléen est égal au Ou logique entre le résultat booléen de l'instruction précédente et l'état inverse de l'opérande.
ORR	—| |— en parallèle avec —|P|—	Le résultat booléen est égal au Ou logique entre le résultat booléen de l'instruction précédente, et la détection d'un front montant de l'opérande (1=front montant).
ORF	—| |— en parallèle avec —|N|—	Le résultat booléen est égal au Et logique entre le résultat booléen de l'instruction précédente et l'état de détection d'un front descendant de l'opérande (1=front descendant).

AND(		Et logique (8 niveaux de parenthèses)
OR(		Ou logique (8 niveaux de parenthèses)
XOR, XORN, XORR, XORF		Ou exclusif
MPS MRD MPP		Aiguillage vers des bobines.
N		Négation*



Instructions d'action

Le tableau suivant décrit les instructions de test du langage liste d'instructions.

Désignation	Graphisme	Fonctions
ST		L'opérande associé prend la valeur du résultat de la zone test.
STN		L'opérande associé prend la valeur inverse du résultat de la zone test.
S		L'opérande associé est mis à 1 lorsque le résultat de la zone test est à 1.
R		L'opérande associé est mis à 0 lorsque le résultat de la zone test est à 1.
JMP		Permet un branchement inconditionnel à une phrase étiquetée, amont ou aval.
JMPC	-	Permet un branchement conditionné à un résultat booléen à 1, à une phrase étiquetée amont ou aval.
JMPCN	-	Permet un branchement conditionné à un résultat booléen à 0, à une phrase étiquetée amont ou aval.
SRn	-	Branchement en début de sous-programme.
RET	-	Retour de sous-programme.
RETC	-	Retour de sous-programme conditionné à un résultat booléen à 1.

RETCN	-	Retour de sous-programme conditionné à un résultat booléen à 0.
END	-	Fin de programme.
ENDC	-	Fin de programme conditionné à un résultat booléen à 1.
ENDCN	-	Fin de programme conditionné à un résultat booléen à 0.

Instruction sur bloc fonction

Le tableau suivant décrit les instructions de test du langage liste d'instructions.

Désignation	Graphisme	Fonctions
Blocs Temporisateur, Compteur, Monostable, Registre, Programmateur cyclique		Pour chacun des blocs fonction standards, il existe des instructions permettant de piloter le bloc. Une forme structurée permet de câbler directement les entrées/sorties des blocs.

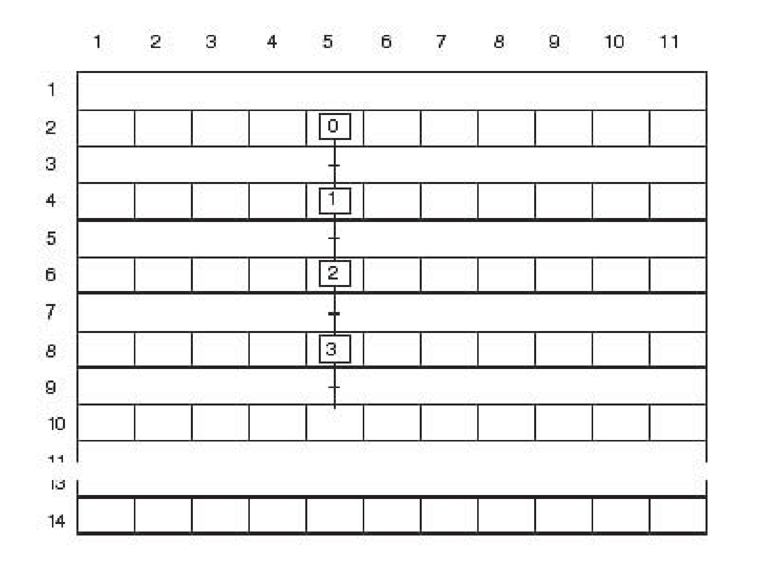
• Représentation du GRAFCET

Généralités

Le graphe principal et chacune des macro-étapes se programment sur 8 pages (page 0 à 7).Une page Grafcet est constituée de 14 lignes et 11 colonnes qui définissent 154 cellules. Dans chaque cellule, il est possible de saisir un élément graphique.

Illustration

Le dessins ci-dessous illustre le découpage d'une page Grafcet.



Règles d'écriture

- La première ligne permet de saisir des renvois d'origine.
- La dernière ligne permet de saisir des renvois de destination.
- Les lignes paires (de 2 à 12) sont des lignes d'étapes (pour les étapes renvois de destination).
- Les lignes impaires (de 3 à 13) sont des lignes de transitions (pour les transitions et les renvois d'origine).
- Chaque étape est repérée par un numéro différent (0 à 127) dans un ordre quelconque.
- Des graphes différents peuvent être représentés sur une même page.

• Possibiliés du GRAFCET

Généralités

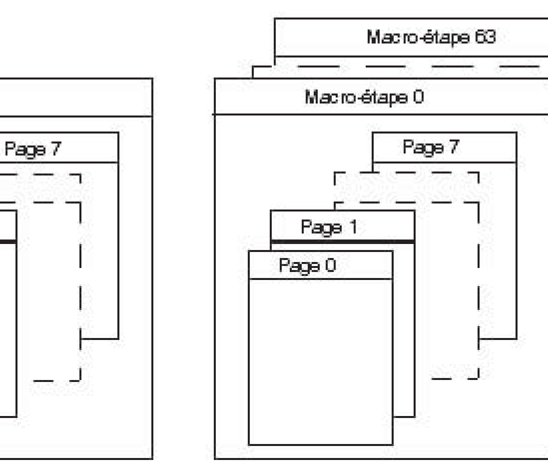
Le traitement séquentiel est structuré en :

- 1 sous ensemble : Graphe principal,
- 64 sous ensembles : Macro-étapes,

Ces sous-ensembles sont eux-mêmes divisés en 8 pages.

Illustration

L'illustration suivante décrit la structure générale en page du Grafcet.



Caractéristiques

Elles dépendent du processeur à programmer, elles sont récapitulées dans le tableau ci-dessous.

Nombre	TSX 37-10		TSX 37-20		TSX 57	
	Par défaut	Maxi	Par défaut	Maxi	Par défaut	Maxi
Etapes du Graphe principal	96	96	128	28	128	250
Macro-étapes	0	0	0	0	8	64
Etapes de macro-étapes	0	0	0	0	64	250
Total d'étapes	96	96	128	128	640	1024
Etapes actives simultanément	16	96	20	128	40	250
Transitions valides simultanément	20	192	24	256	48	400

Le nombre de transitions synchrones (ou nombre de convergences en ET) ne doit pas dépasser 64, le nombre total de transitions étant toujours de 1024.

1- Opérateurs et opérandes

Entrées	24 entrées de **I1 à I24**
Sorties	16 sorties : de **Q1 à Q16**
Variables internes	124 variables internes de **M1** à M124, de M125 à M128 (autres bases de temps)
Temporisateurs	16 temporisateurs :de **T1 à T8** . (base de temps 100ms) , de **T9 à T16** , (base de temps 25ms)
Compteurs	16 compteurs : de **C1 à C16**

Types d'opérations	Opérateurs	Action	Opérandes
Opération logique	A	Opération logique ET, signal positif	I.T.Q.M.C.
	AN	Opération logique ET, signal négatif	I.T.Q.M.C. I.T.Q.M.C.
	O	Opération logique OU, signal positif	I.T.Q.M.C.
	ON	Opération logique OU, signal négatif	
	A(	Opération logique ET, parenthèse ouvert	
	O(	Opération logique OU, parenthèse ouvert	
	)	Parenthèse fermée, signal positif	
	)N	Parenthèse fermée, signal négatif	
Opérations de sorties	=	Sortie positive	Q.M.
	=N	Sortie négative	Q.M.
	SL	Activation mémoire	Q.M.
	RL	Désactivation mémoire	Q.M.
Opération de comptage	=T	Entrée temporisation (sortie temporisations)	
	=Z	Transfert consigne compteur (effacement)	
	=P	Entrée compteur (C: sortie compteur)	
Opérations d'organisation du programme	JI	Saut si « 1 » (Conditionnel positif)	I.Q.
	LS	Chargement immédiat (en mémoire de signaux)	
	NO	Sans effet, opération nulle	
	PE	Fin du programme	

Remarques :

L'activation de l'étape initiale se fait par :

- La mémoire interne M128.
- Le forçage à 1 par un bouton externe.
- Par programmation d'une temporisation .

La programmation se fait par l'activation et la désactivation de chaque étape puis par la programmation des sorties.
Lorsque on écrit un programme en liste d'instructions par exemple, le logiciel transforme automatiquement ce programme en schéma à contact et en logigramme , le travail inverse reste vrai.

2- Programmation d'un GRAFCET : (Liste d'instructions)

Pour écrire la liste d'instructions du programme on doit :

- Etablir le GRAFCET de point de vue PC qui décrit l'automatisme considéré.
- A partir de la table des affectations (Dossier technique) : où on affecte à chaque entrée système une entrée automate (parmi les entrées de I1 à I24), et à chaque sortie système une sortie automate (parmi les sorties de Q1 à Q16).

-Etablir le GRAFCET codé automate (à chaque étape on associe une variable : M1 à M124) .

-Ecrire les instructions du programme formées par :

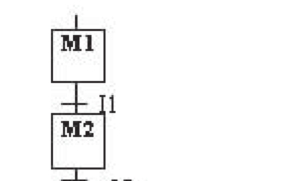
L'initialisation : Au début de chaque programme on écrit les 4 lignes suivantes :

AM128
SLM1 (si M1 est l'étape initiale)
AM128
RLM128

-Le corps du programme :On traduit les équations d'activation et de désactivation de chaque étape par les instructions correspondantes :

Exemple :

Activation de M2 : AM2 = M1. I1==> instructions : **AM 1**
AI1
SLM2



Désctivation de M2 : DM2 = M3 ==> instructions : **AM3**
RLM2

La correspondance entre variables internes et sorties :
exemple : **AM3 = Q2**

-La fin du programme : **PE**

LES MICROCONTRÔLEURS

A- MISE EN SITUATION

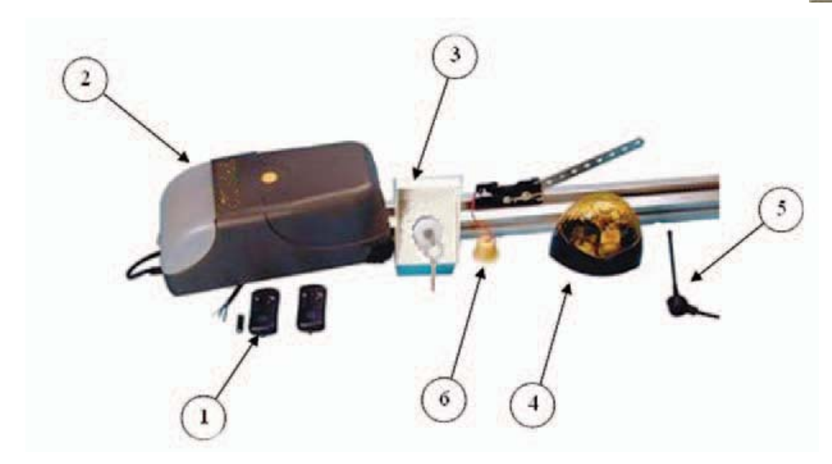
1- Fonctionnement du système : Porte automatique





L'ouverture et la fermeture automatique de la porte d'un garage d'automobile sont assurées par un système technique comportant les éléments suivants :

1. Télécommande à distance.
2. Moteur à courant continu + réducteur.
3. Carte de commande + commutateur à clef.
4. Lampe de signalisation.
5. Antenne de réception.
6. Capteurs de fin de course.

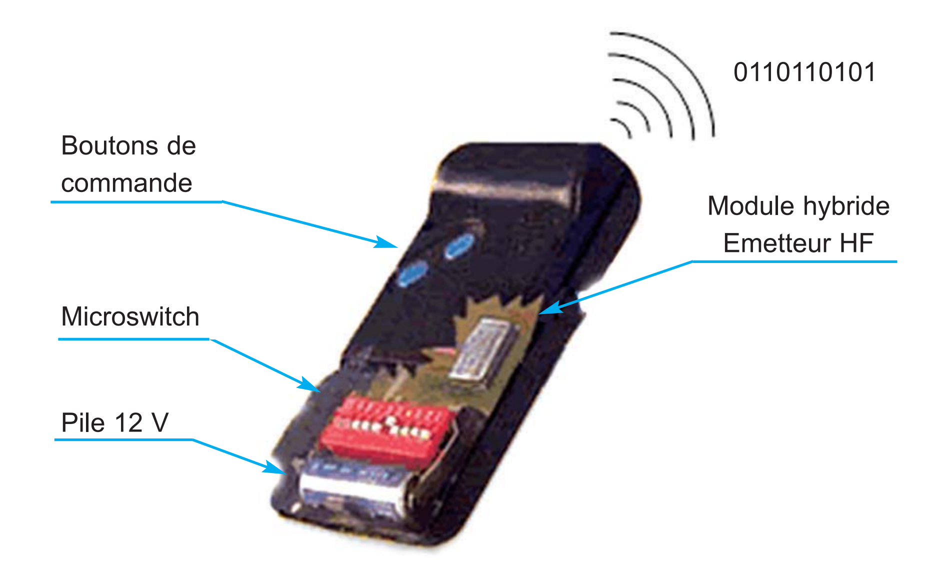


Dans une première partie on va s'intéresser à la télécommande qui nous permet de commander la porte à distance, le principe est basé sur l'émission d'un code binaire sur dix bits **[C9..C0]** reconnaissable par la carte de commande de la porte.

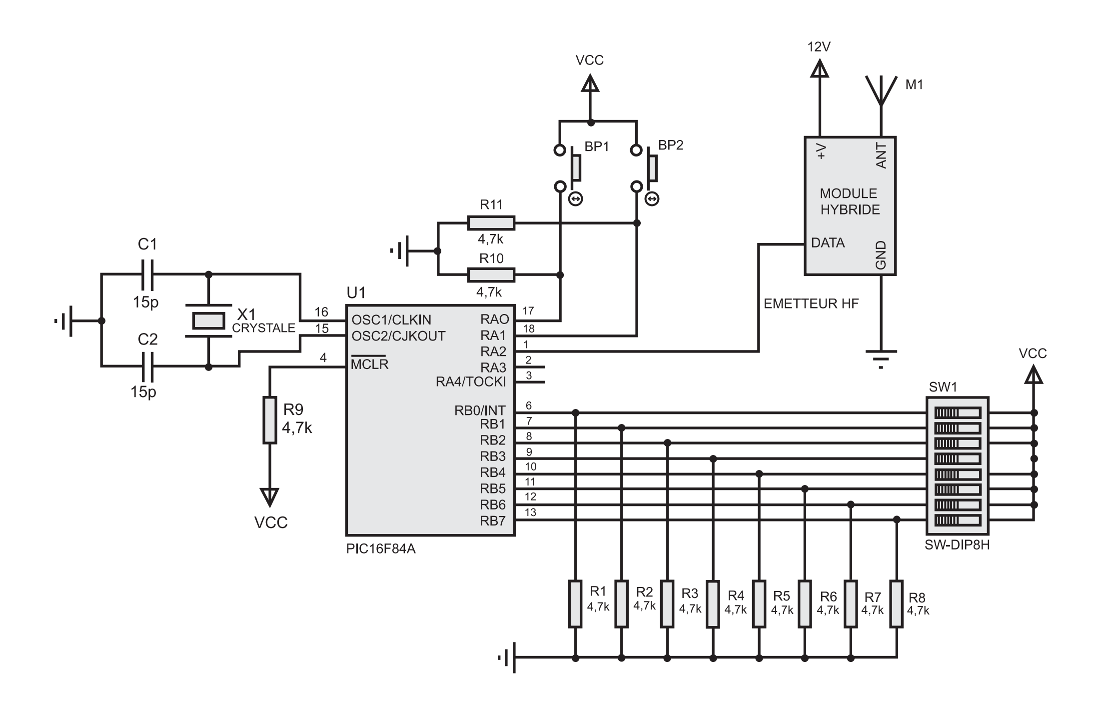
Ce code est composé de deux parties :

Code d'identification de la télécommande								Ordre d'ouverture/fermeture	
C_9	C_8	C_7	C_6	C_5	C_4	C_3	C_2	C_1	C_0

Pour donner la possibilité à l'utilisateur d'adapter une nouvelle télécommande à la porte du garage ou de la remplacer en cas de panne, le constructeur a prévu un **microswitch** sur la télécommande qui nous permet de définir le code d'identification. Celui-ci est généralement inscrit sur la carte de commande ou donné dans le manuel d'utilisation de la porte.
Le code correspondant à l'ordre d'ouverture ou de fermeture de la porte **(C1,C0)** est le suivant : (**01** : ordre d'ouverture ; **11** : ordre de fermeture).



Le schéma de la télécommande est le suivant :



Le module hybride émetteur haute fréquence (M1) est un émetteur sans fil, qui à pour rôle de transmettre le code [C9..C0] généré par le microcontrôleur à la carte de commande de la porte. Ce module ne fera pas l'objet de notre étude.

Activité

Réaliser l'activité N°1 du TP-A41 dans le manuel d'activités

Activité de découverte du système télécommande de porte de garage.



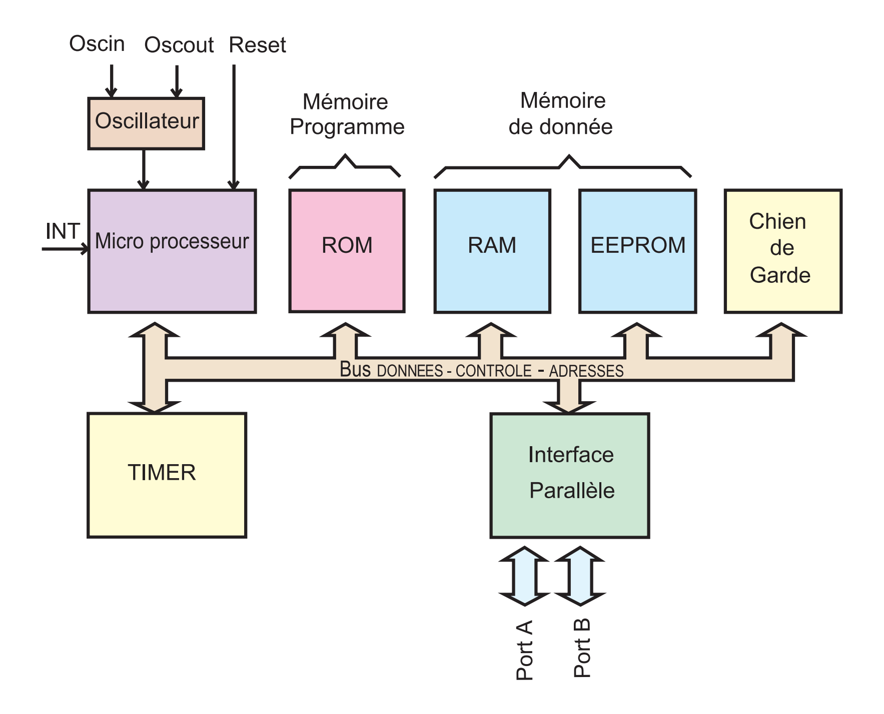
2- Rappel sur les microcontrôleurs :

Un microcontrôleur se présente sous la forme d'un circuit intégré réunissant tous les éléments d'une structure à base de microprocesseur.
Voici généralement ce que l'on trouve à l'intérieur d'un tel composant :

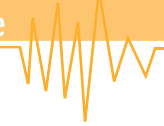
1. Un microprocesseur (C.P.U.).
2. Une mémoire de donnée (RAM et EEPROM).
3. Une mémoire programme (ROM, OTPROM, UVPROM ou EEPROM).
4. Une interface parallèle pour la connexion des entrées / sorties.
5. Une interface série (synchrone ou asynchrone) pour le dialogue avec d'autres unités.
6. Des timers pour générer ou mesurer des signaux avec une grande précision temporelle.
7. Des convertisseurs analogique / numérique pour le traitement des signaux analogiques.

2-1 Microprocesseur (C.P.U.) :

Un microprocesseur exécute séquentiellement les instructions stockées dans la mémoire programme.
On peut noter qu'il existe 2 catégories de microprocesseur : les CISC et les RISC.
CISC (Complex Instruction Set Computer) : Ce microprocesseur possède un nombre important d'instructions. Chacune d'elles s'exécute en plusieurs périodes d'horloges.
RISC (Reduced Instruction Set Computer) : Ce microprocesseur possède un nombre réduit d'instructions. Chacune d'elles s'exécute en une période d'horloge.
Il est généralement construit autour d'une unité arithmétique et logique (ALU) permettant d'effectuer des opérations de calcul arithmétique et des opérations binaires.



Structure interne du microcontrôleur PIC 16F84A



2-2 Mémoires programmes

Ce dispositif contient les instructions du programme que doit exécuter le microprocesseur. Ce type de mémoire est uniquement accessible en lecture. Sa programmation nécessite une procédure particulière et un matériel adéquat.
Il en existe différents types selon leur mode de programmation :

1. ROM dont le contenu est programmé lors de sa fabrication (appelée mémoire morte).
2. PROM programmable électriquement une seule fois par le développeur (appelée aussi OTPROM),
3. EPROM programmable électriquement et effaçable aux U-V (appelée aussi UVPROM),
4. EEPROM ou FLASH programmable et effaçable électriquement.

2-3 Mémoires de données

Ce dispositif permet de mémoriser temporairement les données générées par le microprocesseur pendant les différentes phases du traitement numérique (résultats d'opérations, états logiques...). Ces mémoires sont accessibles en écriture et en lecture.
On en trouve 2 types :
La mémoire (RAM) volatile (données perdues en cas de coupure de l'alimentation) ayant un temps de lecture et écriture assez court (quelques ns),
La mémoire (EEPROM) non-volatile (données conservées en cas de coupure de l'alimentation) ayant un temps d'écriture assez élevé (quelques ms) par rapport au temps de lecture qui est assez faible (quelques ns).

2-4 L'interface parallèle

Ce type d'interface, répartie sur plusieurs ports (maximum 8 bits), permet de prendre en compte des états logiques appliqués en entrée (état de capteurs) ou de ces ports peuvent donc être configurées en entrée ou en sortie, avec différentes options (résistances de rappel, sorties collecteurs ouverts, interruption...). La configuration ainsi que l'état logique de ces broches est obtenue par des opérations d'écriture ou de lecture dans différents registres associés à chaque port. On trouve généralement :
Un registre de direction (TRISX) pour une configuration en entrée ou en sortie du Port X.

Exemple :
On écrit dans le registre TRISB la valeur binaire 10100110, le tableau suivant donne la configuration du port B correspondante.

TRISB	1	0	1	0	0	1	1	0
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
	entrée	sortie	entrée	sortie	sortie	entrée	entrée	sortie

Un registre de donnée (PORTX) recopiant les états logiques de chaque broche du port X. Il est accessible en écriture (sortie) ou en lecture (entrée).
Un registre d'interruption (INTCON) permettant d'activer ou de désactiver les interruptions.



2-5 Le timer

Le Timer permet de réaliser les fonctions suivantes :

1. Génération d'un signal périodique.
2. Temporisation.
3. Comptage d'événements.

Plusieurs registres associés au Timer permettent de configurer les différents modes décrits précédemment.

2-6 Le chien de garde

Ce dispositif est un système anti-plantage du microcontrôleur. Il s'assure qu'il n'y ait pas d'exécution prolongée d'une même suite d'instruction. Si oui un Reset est généré automatiquement relançant ainsi le microcontrôleur.
Il faut donc penser à désactiver le chien de garde lorsqu'il y a des boucles répétitives dans le programme.

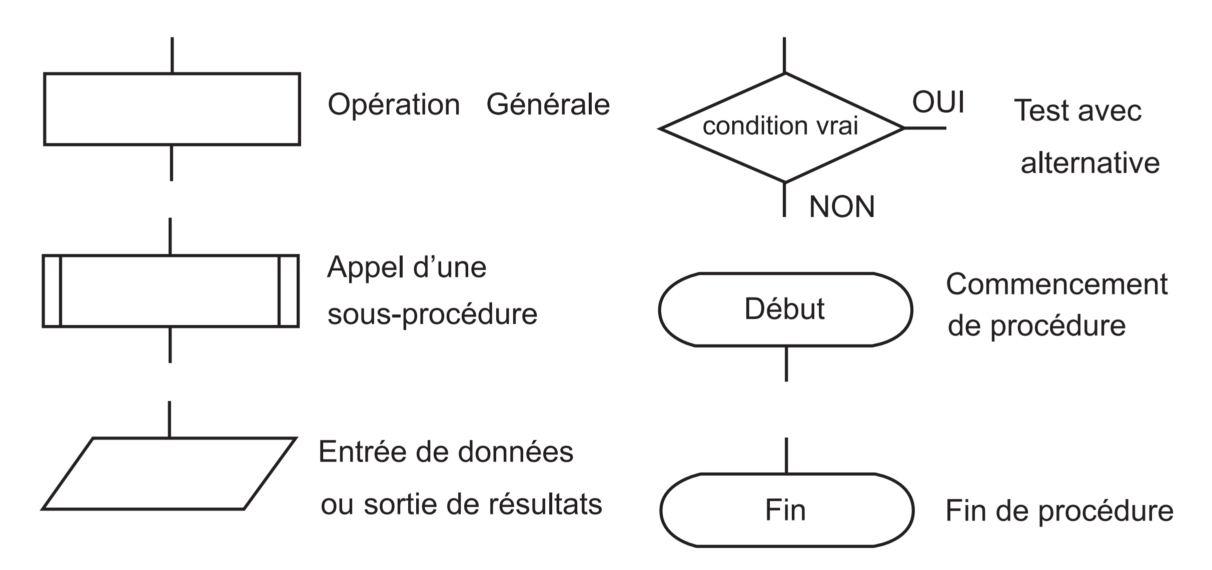
Activité
Réaliser l'activité N°2 du TP-A41 dans le manuel d'activités
Il s'agit d'identifier les différentes caractéristiques du microcontrôleur PIC16F84.

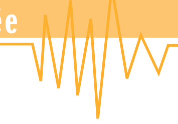
3- Rappel sur la programmation graphique :

3-1 L'algorigramme ou l'organigramme :

Définition : il s'agit d'une représentation graphique et normalisée utilisée pour analyser ou décoder un problème de logique.

Représentation normalisée : il s'agit de dessiner une suite de symboles définie comme suit :





3-2 Application relative à la télécommande de la porte automatique :

Voici le programme du microcontrôleur de la télécommande réalisé par un logiciel graphique :

BIGIN
A — point de connection A
INPUT — Input BP1 <-RA0
INPUT — Input BP2 <-RA1
? — Decision BP1 = 1? — No / Yes
? — Decision BP2 = 1? — No / Yes — A — aller au point A
OUTPUT — Output 1 ->RA2
DELAY — Delay 20ms
appel sous-program TRANS_CODE
OUTPUT — Output 0->RA2
A — aller au point A
END

(Branche Yes de BP1 = 1?)
OUTPUT — Output1 ->RA2
DELAY — Delay 10 ms
OUTPUT — Output 0 ->RA2
DELAY — Delay 10 ms
appel sous-program TRANS_CODE
OUTPUT — Output 0 ->RA2
A — aller au point A

Programme principal

BEGIN
INPUT — Input C <-RB0
OUPUT — Output C ->RA2
DELAY — Delay 10 ms
INPUT — Input C <-RB1
OUTPUT — Output C ->RA2
DELAY — Delay 10 ms
INPUT — Input C <-RB2
OUTPUT — Output C ->RA2
DELAY — Delay 10 ms
INPUT — Input C <-RB3
OUTPUT — Output C ->RA2
DELAY — Delay 10 ms
INPUT — Input C <-RB4
OUPUT — Output C ->RA2
DELAY — Delay 10 ms
INPUT — Input C <-RB5
OUPUT — Output C ->RA2
DELAY — Delay 10 ms
INPUT — Input C <-RB6
OUPUT — Output C ->RA2
DELAY — Delay 10 ms
INPUT — Input C <-RB7
OUPUT — Output C ->RA2
DELAY — Delay 10 ms
END

Sous programme TRANS_CODE

Activité

Réaliser l'activité N°3 du TP-A41 dans le manuel d'activités

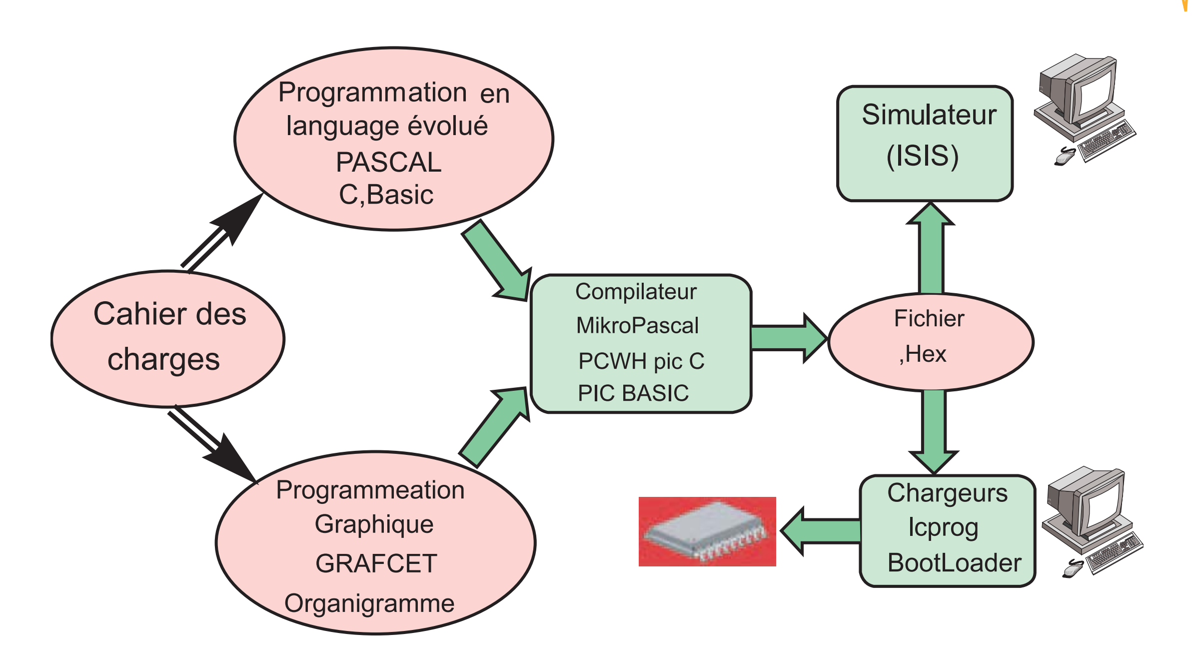
Il s'agit d'exploiter l'organigramme de commande de la télécommande pour dégager le principe de fonctionnement.

4- La programmation en langage évolué :

Que ce soit par la méthode graphique ou en langage évolué, l'écriture du programme ainsi que sa mise au point doivent suivre le diagramme suivant :

Il faut traduire le cahier des charges en une suite ordonnée d'actions que doit réaliser le processus de commande, cette suite d'opérations sera décomposée en actions élémentaires ou instructions c'est l'Algorithme. Par la suite il suffit de transformer cet algorithme en un langage évolué tel que le langage PASCAL ou le langage C.

Dans la suite du cours on s'intéressera au langage PASCAL. (Compilateur Mikropascal de Mikroelektronika.)



Activité

Réaliser l'activité N°4 du TP-A41 dans le manuel d'activités

Il s'agit de lire correctement un algorithme et en déduire l'algorithme de commande de la commande.

4-1 Structure d'un programme :

Un programme est un texte que le compilateur va traduire en fichier hexadécimal. Alors il doit avoir une structure particulière.

Le texte d'un programme contient au moins trois parties.

- L'entête

Ne contient qu'une ligne; commence par le mot réservé " Program " suivi du nom du programme.

- Les déclarations

Elles permettent de définir les éléments utilisés dans le programme.
En effet on devra déclarer les variables utilisées pour permettre au compilateur d'effectuer les réservations de mémoire ainsi que les sous programmes (Procédures et fonctions).

- Le corps du programme

Commence par le mot réservé " Begin " et se termine par le mot réservé "End " suivi d'un point final. Ce qui suit ce "End" n'est pas pris en compte par le compilateur.
Entre "Begin" et "End" se trouvent les instructions à effectuer par le programme.

Algorithmique	Langage PASCAL	
Algorithme NomAlgorithme ;	**Program** NomProgramme ;	Entête
Variables Nomvariable Type **Constantes** Nomconstante Type = valeur	**Var** Nomvariable Type ; **Const** Nomconstante Type = valeur ;	// déclaration
début .... **fin.**	**Begin** .... **End**.	*// Programme* *// principal*

4-2 Les Règles de bases :

- Toutes instructions ou actions se terminent par un point virgule ;
- Une ligne de commentaires doit commencer par "{" et se terminer par "}" ou commence par "//".
- Un bloc d'instructions commence par "Begin" et se termine par "End".

4-3 Les types de variables utilisées en Mikropascal :

Type	Désignation	Taille	Rang
octet	byte	8–bit	0 ⟶ 255
caractère	char	8–bit	0 ⟶ 255
Mot	word	16–bit	0 ⟶ 65535
Octet signé	short	8–bit	-128 ⟶ 127
Entier	integer	16–bit	-32768 ⟶ 32767
Entier long	longint	32–bit	-2147483648 ⟶ 2147483647
Réel	real	32–bit	± 1.17549435082 * 10^{-38}.. ± 6.80564774407 * 10^{-38}
Tableau	Array[n] of type	n éléments	Rang du type
Chaîne de caractères	string[n]	n caractère	0 ⟶ 255

4-4 Les bases du compilateur Mikropascal :

Le décimal :	A=10 ;
L'hexadécimal	A=$0F ; ou A=0x0F ;
Le binaire	A=%11010100 ;

4-5 Les opérateurs arithmétiques et logiques :

Opérateurs arithmétiques		Opérateurs de comparaison		Opérateurs logiques	
Opérateur	*Opération*	*Opérateur*	*Opération*	*Opérateur*	*Opération*
+	Addition	=	Egalité	AND	ET
–	Soustraction	<>	Différent	OR	OU
*	Multiplication	>	Supérieur	XOR	OU exclusif
/	Division	<	Inférieur	NOT	NON
div	Division entière	<=	Inférieur ou égale	SHL	Décalage à gauche
mod	Reste de la division entière	>=	Supérieur ou égale	SHR	Décalage à droite

4-6 Les structures usuelles :

a/ L'affectation : c'est l'action d'attribuer une valeur à une variable.

Langage algorithmique	Langage PASCAL
a←b+c	a:=b+c

b/ Les structures alternatives :

Langage algorithmique	Langage PASCAL
SI condition **ALORS** **DEBUT** *Traitement ;* .... **FINSI ;**	**IF** condition **THEN** **BEGIN** Traitement ; **END ;**
SI condition **ALORS** **DEBUT** *Traitement 1;* .... ; **FIN ;** **SINON** **DEBUT** *Traitement 2;* .... ; **FINSI ;**	**IF** condition **THEN** **BEGIN** *Traitement 1;* .... ; **END ;** **ELSE** **BEGIN** *Traitement 2;* .... ; **END ;**
SELON *expression* *Valeur_1 : action_1 ;* .... *;* *Valeur_n : action_n ;* **autrement** *: action_0 ;* **FINSELON ;**	**CASE** *expression* ***OF*** *Valeur_1 : action_1 ;* .... *Valeur_n : action_n ;* **ELSE** *action_0 ;* **END ;**

c/ Les structures itératives ou répétitives

Langage algorithmique	Langage PASCAL
I: entier; ....; **POUR I** <variant de *valeur initiale*> **JUSQU'A** *valeur finale* **FAIRE** **DEBUT** *Traitement ;* .... **FINFAIRE ;**	I: integer; ....; **FOR** I:=<valeur initiale> To <Valeur finale> **Do** **BEGIN** *Traitement ;* .... **END ;**
TANQUE condition **FAIRE** **DEBUT** *Traitement ;* .... **FINFAIRE;**	**WHILE condition DO** **BEGIN** *Traitement ;* .... **END;**

Activité
Réaliser l'activité N°5 du TP-A41 dans le manuel d'activités
Il s'agit d'écrire un programme en mikropascal pour le microcontrôleur de la télécommande afin de simuler le fonctionnement puis le tester sur plaque d'essai.

4-7 Les procédures et les fonctions :

Une suite d'instructions peut être rassemblée en un bloc qui peut être appelé depuis plusieurs endroits d'un programme. Ceci donne lieu aux notions de sous programme appelé aussi procédures ou fonctions.

- **Procédures**

Ce sont des groupes d'instructions qui vont former une nouvelle instruction simple utilisable dans un programme. En Pascal il faut les définir avant de les utiliser. Ceci se fait en utilisant une structure similaire à celle d'un programme.

Entête

Procedure Identificateur (Param1:Type1, Param2:Type2,...);

Identificateur est le nom de la procédure; Param1, Param2 ... sont des paramètres que le programme fournit à la procédure sous forme de constantes, de variables ou d'expressions; Type1, Type2 ... sont les types de ces paramètres.

Déclarations

Déclarations de constantes, types, variables utilisés à l'intérieur de la procédure

Corps de la procédure

```
Begin
 Instruction1; Instruction2;........
End;
```

Il s'agit des instructions exécutées par le programme à l'appel de la procédure. Une procédure peut appeler d'autres procédures définies avant elle. L'appel d'une procédure se fait en écrivant son nom suivi des paramètres nécessaires entre parenthèses.

Fonctions

Une fonction est une procédure qui devra fournir un résultat de type numérique ou chaîne de caractères. La définition se fait en utilisant une structure similaire à celle de la procédure.

Entête

Function Identificateur (Param1:Type1, Param2:Type2,...):Type_R;

Identificateur est le nom de la procédure; Param1, Param2 ... sont des paramètres que le programme fournit à la fonction sous forme de constantes, de variables ou d'expressions; Type1, Type2 ... sont les types de ces paramètres; Type_R est le type du résultat fourni par la fonction.

Déclarations

Déclarations de constantes, types, variables utilisés à l'intérieur de la fonction.

Corps de la fonction

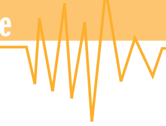
Begin
Instruction1; Instruction2;……..
Identificateur:=résultat;
End;

Il s'agit des instructions exécutées par le programme à l'appel de la fonction. L'une de ces instructions doit fournir le résultat de la fonction en l'affectant au nom de la fonction. L'appel d'une fonction se fait en écrivant son nom suivi des paramètres nécessaires entre parenthèses. Elle représente une expression du type du résultat fourni.

4-8 Les fonctions adaptées aux microcontrôleurs PIC :

Le compilateur mikropascal apporte une large bibliothèque de procédures et fonctions adaptées aux microcontrôleurs de la famille PIC de MICROCHIP. Ces fonctions sont accessibles dans l'aide du logiciel néanmoins on va citer quelque-une.

Fonctions / Procédures	Exemple
Setbit (port, bit) ;	Setbit (portB, 2) ; mettre la broche RB2 à 1
Clearbit (port, bit)	Clearbit (portB, 5) ; mettre la broche RB5 à 0
Testbit (port, bit)	A:=testbit (portB, 7) ; affecter à la variable A l'état de RB7
Delay_ms (temps)	Delay_ms (150) ; attente de 150 ms
Button (port, bit, temps d'appui, état logique actif).	If Button(portA,2,10,1) then <Action 1> ; On teste l'appui sur un bouton poussoir relié à la broche RA2 pendant 10ms pour faire l'Action 1.



4-9 Application :

La carte de commande de la porte automatique est aussi construite à base du microcontrôleur PIC16F84A. Elle contient les éléments suivants :

- Un commutateur à clef permettant à l'utilisateur d'ouvrir la porte en manoeuvrant une clef.
- Un module récepteur hybride HF permettant la réception d'ordre d'ouverture ou de fermeture de la télécommande à distance.
- Des relais pour la commande du moteur et de la lampe.

Voici le schéma interne de la carte de commande :

Activité

Réaliser l'activité N°6 du TP-A41 dans le manuel d'activités

Il s'agit d'écrire un programme en mikropascal afin de simuler le fonctionnement de la carte de commande de la porte du garage.

5- Les interruptions

5-1 Qu'est-ce qu'une interruption ?



Imaginez une conversation normale. Chaque interlocuteur prend la parole quand vient son tour de parler. Survient alors un évènement extérieur dont le traitement est urgent. Par exemple une voiture s'engage dans votre direction. Vous imaginez bien que votre interlocuteur ne va pas attendre la fin de votre phrase pour vous signaler le danger. Il va donc vous interrompre durant le cours normal de votre conversation ; afin de pouvoir traiter immédiatement l'événement extérieur. Les interlocuteurs reprendront leur conversation où elle en était arrivée, sitôt le danger écarté.

5-2 Les interruptions du microcontrôleur PIC16F84 :

L'exécution d'une procédure d'interruption répond à un évènement qui peut être interne ou externe au microcontrôleur. L'appel d'une procédure d'interruption ne dépend pas du programme principal, mais elle l'interrompe pendant son exécution.
Dans une procédure d'interruption on ne peut pas appeler une autre procédure. Le compilateur Mikropascal utilise une directive spéciale pour le nom de la procédure (Procedure interrupt) pour la différencier des autres procédures logicielles.

Syntaxe :

```
Procedure interrupt;
Begin
      Instruction 1;
      .................
      Instruction n;
End;
```

On notera que le 16F84 possède 4 sources d'interruptions:

- Un changement d'état sur les broches RB4 à RB7;
- Une source externe via la broche RB0/INT ;
- Le débordement du timer 0 ;
- La programmation de l'EEPROM de données.

Le registre INTCON est utilisé pour le contrôle et la gestion des interruptions.

INTCON:	GIE	EEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF

On s'intéressera uniquement à l'interruption externe via la broche **RB0/INT**, donc on aura à manipuler uniquement les trois bits suivants :

- Le bit **GIE**, "Global Interrupt Enable", mis à 1 autorise toutes les interruptions non masquées par leur bit individuel.
- Le bit **INTE**, "Interrupt Enable", mis à 1, autorise les interruptions sur **RB0/INT**.
- Le bit **INTF**, "Interrupt Flag", est un indicateur, il est mis à 1 si une interruption est générée sur **RB0/INT**. **Il doit être remis à 0 par le logiciel dans le programme de traitement de l'interruption.**

Les autres bits du registre INTCON sont utilisés pour le contrôle des trois autres sources d'interruptions, dans notre cas ces bits sont égales à 0.
Pour activer l'interruption externe via la broche RB0/INT on doit affecter la valeur (90)H au registre INTCON :

INTCON:	1	0	0	1	0	0	0	0	$=(90)_H$

Pour plus d'information sur les autres sources d'interruptions consultez le document constructeur du microcontrôleur 16F84.
Pour plus d'information sur La programmation en langage évolué consultez le site web :
www.technologuepro.com

Activité

Réaliser l'activité N°7 du TP-A41 dans le manuel d'activités

Il s'agit d'introduire une procédure d'interruption dans le programme du microcontrôleur de la carte de commande de la porte et en fin simuler le fonctionnement de l'ensemble télécommande + carte de commande

B- RÉSUMÉ : Structure générale d'un programme en Mikropascal pour le microcontôleur 16F84A.

Program nom_programme ;
→ *Déclarations de constantes, types, variables utilisés dans le programme.*
→ *Déclarations des procédures et fonctions utilisées dans le programme.*
Begin
→ *Activation des interruptions utilisées*
→ *Configuration des entrées/sorties.*
→ *Initialisation des sorties utilisées.*

```
while (1=1) do
   begin
      Instruction1;
      Instruction2;
      ……..
   end ;
end.
```

→ Déclaration des variables et des constantes :

Var nomvariable : type ;
Const nomconstante : type = valeur ;

→ Déclaration d'une procédure:

Procedure Identificateur (Param1:Type1, Param2:Type2,...);
Déclarations de constantes, types, variables utilisés à l'intérieur de la procédure

```
 Begin
    Instruction1;
    Instruction2;……..
 End;
```

→ **Déclaration d'une fonction :**

Function Identificateur (Param1:Type1, Param2:Type2,...):Type_R;
Déclarations de constantes, types, variables utilisés à l'intérieur de la fonction.
```
 Begin
    Instruction1;
    Instruction2;........
    Identificateur:=résultat;
 End;
```

→ **Déclaration d'une procédure d'interruption :**

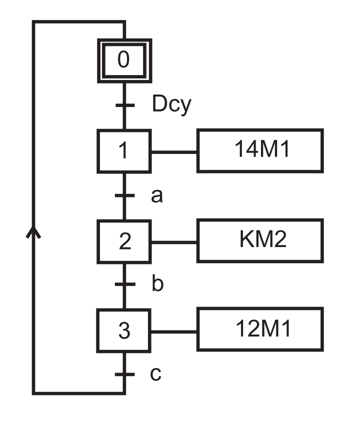
```
Procedure interrupt;
  Begin
       Instruction 1;
       ..................;
       Instruction n;
```
Mise à zéro de l'indicateur correspondant à la source d'interruption utilisée.
```
  End;
```

1- Exercices résolus :

Exercice 1 :

Traduction d'un GRAFCET du point de vue de la partie commande en un programme en Mikropascal pour le micro-contrôleur PIC16F84A.

- Soit le GRAFCET suivant :



- On vous donne le tableau d'affectation des entrées/sorties pour le microcontrôleur PIC16F84A.

Entrées		Sorties	
Capteurs	Broches du microcontrôle	Pré-actionneur	Broches du microcontrôleur
Dcy	RB0	14M1	RA0
a	RB1	KM2	RA1
b	RB2	12M1	RA2
c	RB3		



Schéma de simulation :

Travail demandé :

1- Traduire le GRAFCET précédent en algorithme.
2- Transformer cet algorithme en programme en Mikropascal.

Solution :

1- L'algorithme :

```
Algorithme grafcet1 ;
    Variables    X0, X1, X2, X3 : octet ;          // Déclaration des étapes du GRAFCET.
 _ DEBUT
|    TrisA ⇐ $F8 ;    // RA0, RA1, RA2  sont des sorties ; les autres broches sont des entrées.
|    TrisB ⇐ $FF ;                 // Toutes  les broches du  port B sont des entrées.
|    PortA ⇐ 0 ;                                        // état initial des sorties.
|    X0 ⇐ 1 ;                               // Initialement l'étape " X0 " est active.
|    X1 ⇐ 0 ;                               // Initialement l'étape " X1 " est non active.
|    X2 ⇐ 0 ;                               // Initialement l'étape " X2 " est non active.
|    X3 ⇐ 0 ;                               // Initialement l'étape " X3 " est non active.
|   TANQUE (1=1) FAIRE                                              // Boucle infinie.
|        _ DEBUT
|       |  SI ((X0 =1) ET (Portb.0=1)) ALORS          // Condition de franchissement de la
|       |                                            // première transition qui est :
|       |                                    // étape " X0 "  active et réceptivité " Dcy " vraie.
|       |                          _ DEBUT
|       |                         |    X0⇐ 0 ;              // Désactivation de l'étape " X0 ".
|       |                         |    X1⇐ 1 ;                  // Activation de l'étape " X1 ".
|       |                         |_ FINSI ;
|       |  SI ((X1 =1) ET (Portb.1=1)) ALORS          // Condition de franchissement de la
|       |                                            // deuxième transition qui est :
|       |                                    // étape " X1 "  active et réceptivité " a " vraie.
|       |                          _ DEBUT
|       |                         |    X1⇐ 0 ;              // Désactivation de l'étape " X1 ".
|       |                         |    X2⇐ 1 ;                // Activation de l'étape  "  X2 ".
|       |                         |_ FINSI ;
|       |  SI ((X2 =1) ET (Portb.2=1)) ALORS          // Condition de franchissement de la
|       |                                            // troisième transition qui est :
|       |                                    // étape " X2 "  active et réceptivité  " b " vraie.
|       |                           _ DEBUT
|       |                          |     X2⇐ 0 ;            // Désactivation de l'étape " X2 ".
|       |                          |     X3⇐ 1 ;                // Activation de l'étape " X3 ".
|       |                          |_ FINSI ;
|       |  SI ((X3 =1) ET (Portb.3=1)) ALORS          // Condition de franchissement de la
|       |                                            // quatrième transition qui est :
|       |                                    // étape " X3 " active et réceptivité " c " vraie.
|       |                          _DEBUT
|       |                          |     X3⇐ 0 ;            // Désactivation de l'étape " X3 ".
|       |                          |     X0⇐ 1 ;                // Activation de l'étape " X0 ".
|       |                          |_FINSI ;
|       |  SI (X1=1) ALORS Porta.0 ⇐1 SINON Porta.0 ⇐0 ; FINSI;      // 14M1 = X1.
|       |  SI (X2=1) ALORS Porta.1 ⇐1 SINON Porta.1 ⇐0 ; FINSI;      // KM2 = X2.
|       |  SI (X3=1) ALORS Porta.2 ⇐1 SINON Porta.2 ⇐0 ; FINSI;      // 12M1 = X3.
|       |_ FINFAIRE ;
|_  FIN.
```

2- Le programme :

```
program GRAFCET1;
Var  X0,X1,X2,X3:byte ;                              // Déclaration des étapes du GRAFCET.
begin
TrisA := $F8 ;        // RA0,RA1,RA2 sont des sorties ; les autres broches sont des entrées.
TrisB := $FF ;                        // Toutes  les broches du  port B sont des entrées.
PortA:= 0 ;                                                // état initial des sorties.
 X0:=1;                                       // Initialement l'étape " X0 " est active.
 X1:=0;                                   // Initialement l'étape " X1 " est non active.
 X2:=0;                                   // Initialement l'étape " X2 " est non active.
 X3:=0;                                   // Initialement l'étape " X3 " est non active.
 while (1=1) do                                                     // Boucle infinie.
  begin
  if ((X0 =1) and (Portb.0=1)) then               // Condition de franchissement de la
                                                  // première transition qui est :
                                      // étape " X0 "  active et réceptivité " Dcy " vraie.
             begin
             X0:=0;                                   // Désactivation de l'étape " X0 ".
             X1:=1;                                      // Activation de l'étape " X1 ".
             end ;
 if ((X1 =1) and (Portb.1=1)) then                // Condition de franchissement de la
                                                    // deuxième transition qui est :
                                        // étape " X1 "  active et réceptivité " a " vraie.
             begin
             X1:=0;                                   // Désactivation de l'étape " X1 ".
             X2:=1;                                    // Activation de l'étape  "  X2 ".
             end ;
  if ((X2 =1) and (Portb.2=1)) then               // Condition de franchissement de la
                                                   // troisième transition qui est :
                                       // étape " X2 "  active et réceptivité  " b " vraie.
              begin
              X2:=0;                                  // Désactivation de l'étape " X2 ".
              X3:=1;                                     // Activation de l'étape " X3 ".
              end ;
   if ((X3 =1) and (Portb.3=1)) then              // Condition de franchissement de la
                                                    // quatrième transition qui est :
                                         // étape " X3 " active et réceptivité " c " vraie.
              begin
              X3:=0 ;                                 // Désactivation de l'étape " X3 ".
              X0:=1 ;                                       // Activation de l'étape " X0 ".
              end ;
   if (X1=1) then Porta.0:=1  else Porta.0:= 0 ;                          // 14M1 = X1.
   if (X2=1) then Porta.1:=1  else Porta.1:= 0 ;                          // KM2 = X2.
   if (X3=1) then Porta.2:=1  else Porta.2:= 0 ;                          // 12M1 = X3.
 end;
 end.
```



Exercice 2 :

Traduction des équations d'un système combinatoire en un programme en Mikropascal pour le microcontrôleur PIC16F84A.

Soit les équations suivantes :

$S1 = A \cdot B \cdot \overline{C}$

$S2 = B \oplus C$

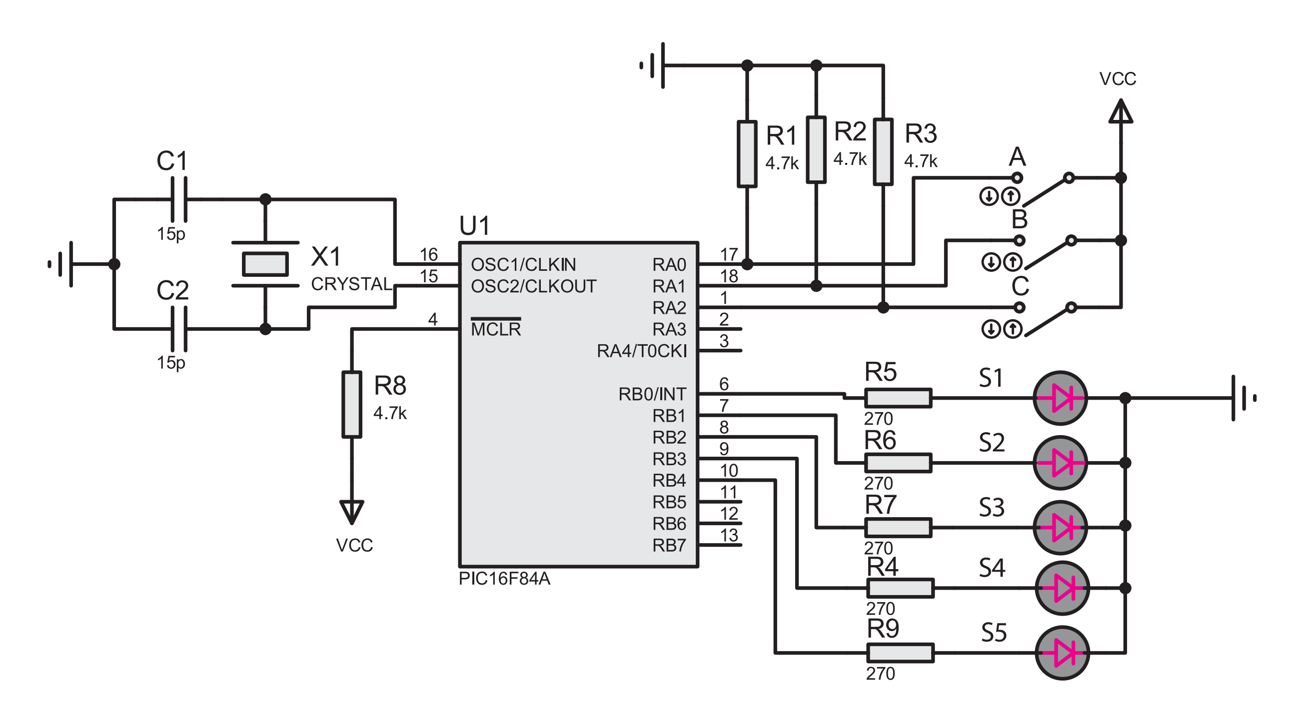
$S3 = A + B \cdot \overline{C}$

$S4 = \overline{A + B}$

$S5 = \overline{B \cdot C}$

Affectation des entrées/sorties pour le PIC16F84A			
Entrées		**Sortie**	
A	RA0	S1	RB0
B	RA1	S2	RB1
C	RA2	S3	RB2
		S4	RB3
		S5	RB4

Schéma de simulation :



Travail demandé :

1- Traduire les équations précédentes en algorithme.
2- Transformer cet algorithme en programme en Mikropascal.

Solution :

Première méthode :

1- Algorithme :

```
Algorithme  EQUATIONS1;
_DEBUT
| trisA ⇐ $FF;                  // Tout le port A est configuré en entrée .
| trisB ⇐ $E0;        // de RB0 à RB4 sorties les autres broches : entrées .
| portb ⇐ 0;                    // initialisation des sorties
| TANQUE  (1=1)  FAIRE          // boucle infinie
|   _DEBUT
|   |// équation de S0 :
|   |  SI ((portA.0=1) ET (portA.1=1) ET ( portA.2=0)) ALORS
|   |                      portb.0⇐1 SINON portb.0 ⇐ 0; FINSI ;
|   |//équation de S1 :
|   |  SI ((portA.1=1) ⊕ (portA.2=1)) ALORS
|   |                      portb.1⇐1 SINON portb.1 ⇐ 0; FINSI ;
|   |//équation de S2 :
|   |  SI  ((portA.0=1) OU ((portA.1=1) ET (portA.2=0))) ALORS
|   |                      portb.2⇐1 SINON portb.2 ⇐ 0; FINSI ;
|   |// équation de S3
|   |  SI  ((portA.1=0) ET (portA.2=0)) ALORS
|   |                      portb.3⇐1 SINON portb.3⇐0; FINSI ;
|   |// équation de S4
|   |  SI ((portA.1=0) OU (portA.2=0)) ALORS
|   |                      portb.4⇐1 SINON portb.4⇐0; FINSI ;
|   | _FINFAIRE;
|_FIN.
```

2- Programme :

```
program EQUATIONS1;
begin
trisA:=$FF;                     // Tout le port A est configuré en entrée .
trisB:=$E0;                     // de RB0 à RB4 sorties les autres broches : entrées.
portb:=0;                       // initialisation des sorties
while (1=1) do                  // boucle infinie
    begin
    // équation de S0 :
    if((portA.0=1) and (portA.1=1) and ( portA.2=0)) then portb.0:=1 else portb.0:=0;
    //équation de S1 :
    if((portA.1=1) xor (portA.2=1))then portb.1:=1 else portb.1:=0;
    //équation de S2 :
     if ((portA.0=1) or ((portA.1=1) and (portA.2=0))) then portb.2:=1 else portb.2:=0;
    // équation de S3
    if ((portA.1=0) and (portA.2=0)) then portb.3:=1 else portb.3:=0;
    // équation de S4
    if((portA.1=0) or (portA.2=0))then portb.4:=1 else portb.4:=0;
    end;
end.
```

Deuxième méthode

1- Algorithme :

```
Algorithme  EQUATIONS;
Variables
      S1, S2, S3, S4, S5 : octet;                    // déclaration des variables de sortie.
 _ DEBUT
| TrisA ⇐ $FF;                                                      // port A : entrée
| TrisB ⇐ $E0;              //RB0, RB1, RB2, RB3, RB4 : sorties ;  RB5 …RB7 : entrées
| Portb ⇐ 0;                                                  // initialisation des sortie
| TANQUE (1=1)  FAIRE                                               // boucle infinie
|       _ DEBUT
|     |   S1⇐((portA.0) ET (portA.1) ET ((not portA.2)));           // équation de S1
|     |   S2⇐((portA.1) OU exclusif  (portA.2));                     // équation de S2
|     |   S3⇐((portA.0) OU ((portA.1) ET ((not portA.2))));         // équation de S3
|     |   S4⇐ non ((portA.1) OU (portA.2));                          // équation de S4
|     |   S5⇐ non ((portA.1) ET (portA.2));                          // équation de S5
|     |
|     |   S1⇐S1 ET $01;                 // S1, S2, S3, S4, S5 sont de type octet (8 bits)
|     |   S2⇐S2 ET $01;                 //  or on veut travailler avec un seul bit
|     |   S3⇐S3 ET $01;                 // qui est le poids le plus faible (2^0)
|     |   S4⇐S4 ET $01;                 // alors il faut masquer les autres bits.
|     |   S5⇐S5 ET $01;                 // exp a (ocet) ; si a=0  alors non a=(11111111)2
|     |
|     |   SI (S1=1) ALORS portb.0⇐1 SINON portb.0⇐0; FINSI;          // RB0 = S1.
|     |   SI (S2=1) ALORS portb.1⇐1 SINON portb.1⇐0; FINSI;          // RB1 = S2.
|     |   SI (S3=1) ALORS portb.2⇐1 SINON portb.2⇐0; FINSI;          // RB2 = S3.
|     |   SI (S4=1) ALORS portb.3⇐1 SINON portb.3⇐0; FINSI;          // RB3 = S4.
|     |   SI (S5=1) ALORS portb.4⇐1 SINON portb.4⇐0; FINSI;          // RB4 = S5.
|     |_FINFAIRE;
|_FIN.
```

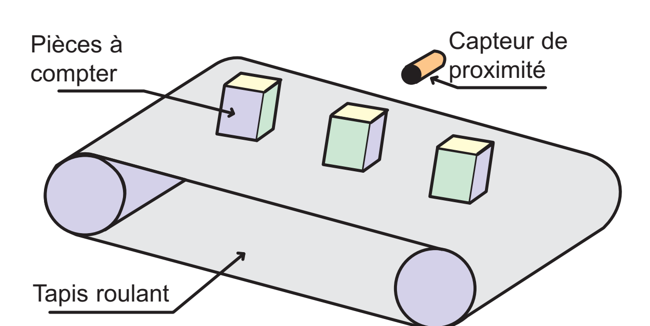
2- Programme :

```
program EQUATIONS;
Var S1,S2,S3,S4,S5 : byte;                       // déclaration des variables de sortie.
begin
trisA:=$FF;                                      // port A : entrée.
trisB:=$E0;                 //RB0, RB1, RB2, RB3, RB4 : sorties ;  RB5 …RB7 : entrées
portb:=0;                                                 // initialisation des sortie
while (1=1) do                                                      // boucle infinie
    begin
    S1:=((portA.0) and (portA.1) and ((not portA.2)));              // équation de S1
    S2:=((portA.1) xor (portA.2));                                  // équation de S2
    S3:=((portA.0) or ((portA.1) and ((not portA.2))));             // équation de S3
    S4:= not((portA.1) or (portA.2));                               // équation de S4
    S5:= not((portA.1) and (portA.2));                              // équation de S5
    S1:=S1 and $01;   S2:=S2 and $01;              //  or on veut travailler avec un seul bit
```

```
    S3:=S3 and $01;                    // qui est le poids le plus faible (2^0)
    S4:=S4 and $01;                    // alors il faut masquer les autres bits.
    S5:=S5 and $01;                    // exp a (ocet) ; si a=0  alors non a=(11111111)2
    if (S1=1) then portb.0:=1 else portb.0:=0;          // RB0 = S1.
    if (S2=1) then portb.1:=1 else portb.1:=0;          // RB1 = S2.
    if (S3=1) then portb.2:=1 else portb.2:=0;          // RB2 = S3.
    if (S4=1) then portb.3:=1 else portb.3:=0;          // RB3 = S4.
    if (S5=1) then portb.4:=1 else portb.4:=0;          // RB4 = S5.
    end;
end.
```

Exercice 3 :

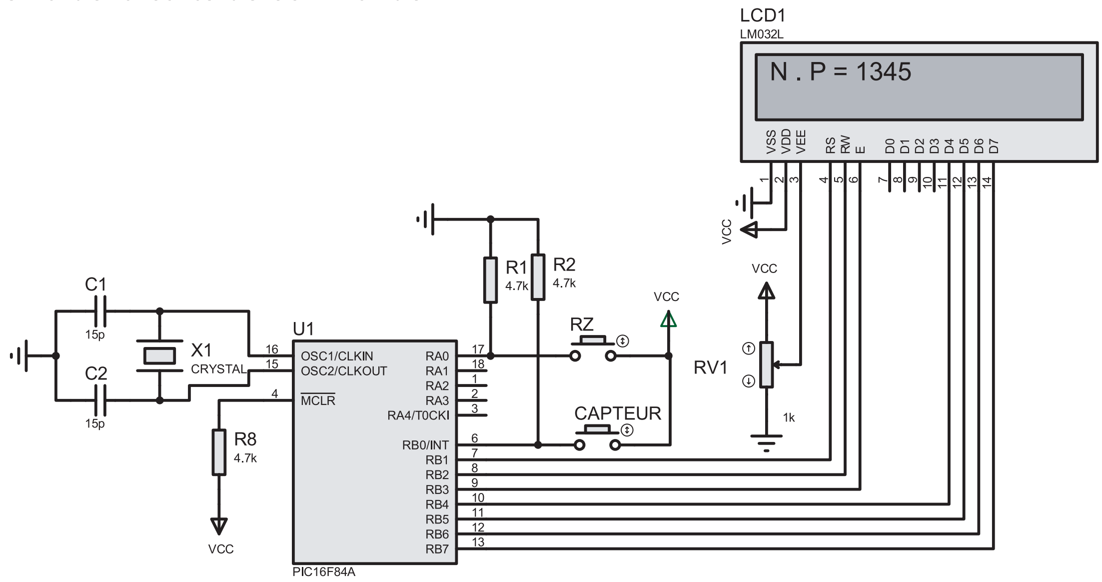
Utilisation de l'interruption externe pour le comptage de pièces sur un tapis roulant dans un processus industriel.



Principe de fonctionnement :

- Ce tapis roulant se trouve à la fin d'une chaine de fabrication de pièces mécaniques.
- Un capteur de proximité permet de compter le nombre de pièces fabriquées.
- Un bouton poussoir RZ permet de mettre le compteur à zéro.
- Le nombre de pièce comptées est affiché par un afficher LCD.

Schéma de la carte de commande :



Travail demandé :

1- Traduire le fonctionnement de ce système par algorithme.

2- Transformer cet algorithme en programme en Mikropascal.

Solution :

1- L'algorithme :

```
Algorithme COMPTEUR;
Variables
      compteur : entier;  // déclaration de la variable de comptage : entier.
         nombre : chaine de caractère [12];  // déclaration d'une chaine de 12 caractères
                                                          // pour l'affichage.
procédure interruption;     // sous programme d'interruption.
 _DEBUT
 |  intcon⇐$90;    // validation de l'interruption RB0 et mise à zéro de son drapeau le bit INTF
 |  compteur⇐compteur+1; // incrémentation du compteur.
 |_FIN;

_DEBUT
|    lcd_config(portb,1,3,2,7,6,5,4);          //configuration de l'LCD doit toujours être avant la
|                                                //configuration des registres TRISx
|                                               // lcd_config(le port, RS, E, RW, D7, D6, D5, D4)
|    lcd_cmd(lcd_cursor_off);                           // éteindre le curseur de l'afficheur LCD
|    TrisA⇐$FF;                           //portA entrée
|    TrisB⇐$01;                         // RB0 : entrée ; les autres broches : sorties pour l'LCD
|    intcon⇐$90;                                  // activation de l'interruption externe RB0/INT
|    compteur⇐0;   // initialisation du compteur.
|     lcd_out(1,1,'N.P =');  //afficher sur la 1ere ligne et la 1ere colonne de l'LCD " N.P= ".
|    TANQUE  (1=1)  FAIRE // boucle infinie.
|          _DEBUT
|          |    inttostr (compteur,nombre);          // changement de la variable " compteur "
|          |                                          //en une chaine de caractère qui est " nombre ".
|          |   lcd_out(1,6,nombre);            //afficher sur la 1ere ligne
|          |                                  //et la 6ème  colonne de l'LCD :" nombre ".
|          |  SI (porta.0=1) ALORS compteur⇐0;FINSI;          // l'appui sur le bouton RZ
|          |                                                  // initialise le compteur.
|          |_FINFAIRE;
|_FIN.
```

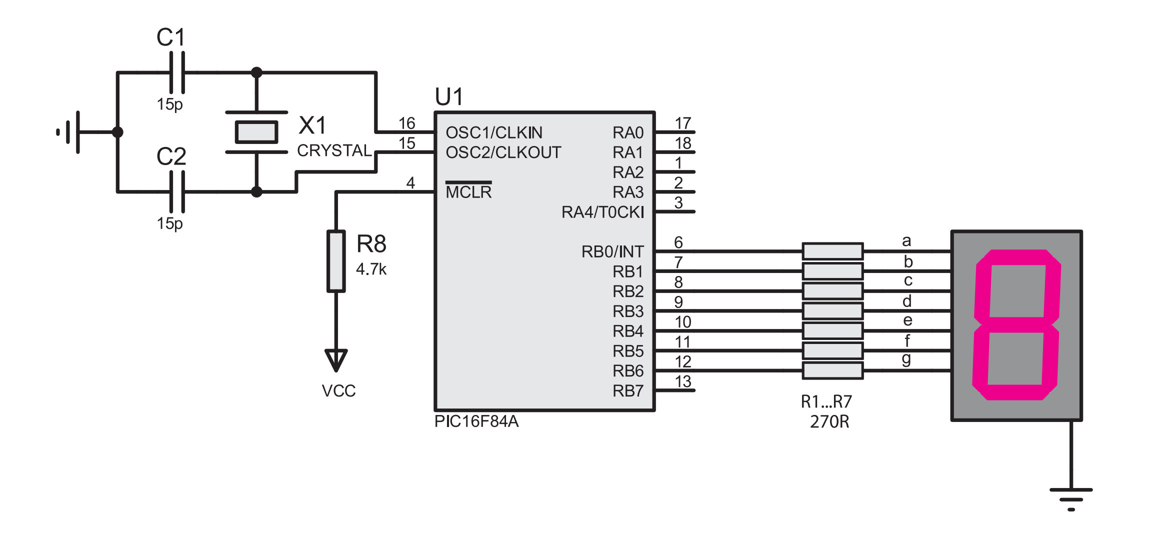
2- Le programme :

```
program COMPTEUR;
Var compteur : integer;                    // déclaration de la variable de comptage : entier.
        nombre : string[12];            // déclaration d'une chaine de caractère pour l'affichage.
procedure interrupt;                   // sous programme d'interruption.
        begin
        intcon:=$90;     // validation de l'interruption RB0 et mise à zéro de son drapeau le bit INTF
        compteur:=compteur+1;                                // incrémentation du compteur.
        end;
```

```
begin
lcd_config(portb,1,3,2,7,6,5,4);          //configuration de l'LCD doit toujours être avant la
                                           //configuration des registres TRISx
                                          // lcd_config(le port, RS, E, RW, D7, D6, D5, D4)
lcd_cmd(lcd_cursor_off);              // éteindre le curseur de l'afficheur LCD
TrisA:=$FF;                               //portA entrée
TrisB:=$01;                               // RB0 : entrée ; les autres broches : sorties pour l'LCD
intcon:=$90;                              // activation de l'interruption externe RB0/INT
compteur:=0;                              // initialisation du compteur.
lcd_out(1,1,'N.P =');                 //afficher sur la 1ere ligne et la 1ere colonne de l'LCD " N.P= ".
while (1=1) do                        // boucle infinie.
        begin
        inttostr(compteur,nombre);            // changement de la variable " compteur "
                                                //en une chaine de caractère qui est " nombre ".
        lcd_out(1,6,nombre);                     //afficher sur la 1ere ligne
                                                  //et la 6emme  colonne de l'LCD :" nombre ".
        if (porta.0=1) then compteur:=0;      // l'appui sur le bouton RZ initialise le compteur.
        end;
end.
```

Exercice 4 : Extension d'un port pour le microcontrôleur PIC16F84A.
On désire réaliser un compteur modulo 100 en utilisant le microcontrôleur 16F84A et deux afficheurs sept segments.

A- Etude préliminaire :

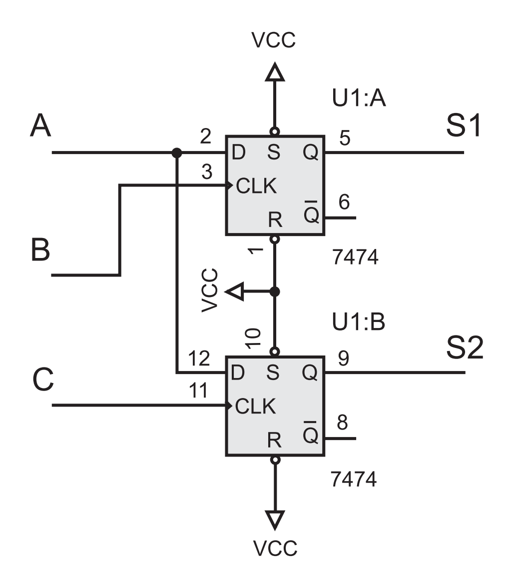


On vous donne le schéma suivant :

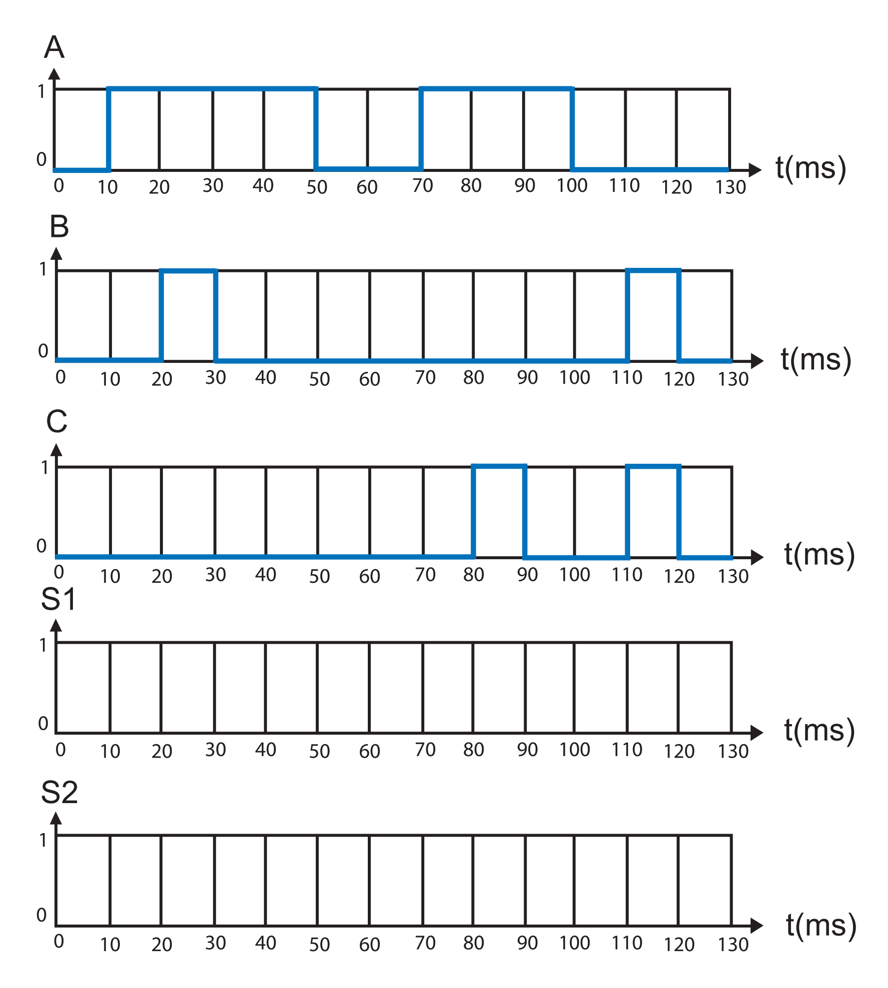
A-1 Ecrire le programme correspondant afin de réaliser un compteur modulo 10.
A-2 Peut-on ajouter un deuxième afficheur sur les ports du microcontrôleur.



A-3 Soit le schéma suivant :



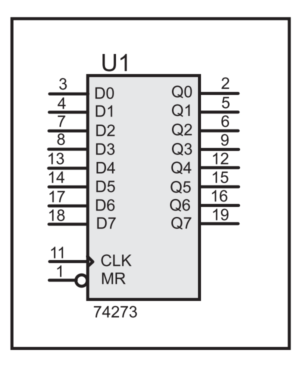
Compléter les chronogrammes suivant :



B- Extension du port B du microcontrôleur :

B-1 On vous donne le circuit 74273, qui contient 8 bascules D avec horloge et remise à zéro communes, proposer une solution pour réaliser un compteur modulo 100 avec le PIC16F84A.

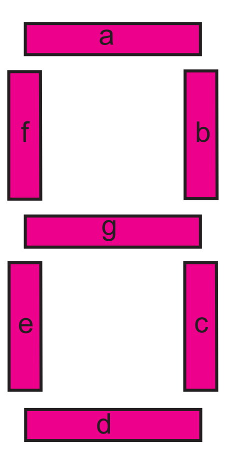
B-2 Ecrire le programme correspondant.



Solution :

A-1 ***Première méthode :***

```
program COMPTEURMOD10a;
begin
Trisb:=0;
portb:=0;
while (1=1) do
begin
//segments [ g f e d c b a]
portb:=%00111111 ; // chiffre 0
delay_ms(1000); // attente d'une seconde
portb:=%00000110; // chiffre 1
delay_ms(1000);
portb:=%01011011; // chiffre 2
delay_ms(1000);
portb:=%01001111; // chiffre 3
delay_ms(1000);
portb:=%01100110; // chiffre 4
delay_ms(1000);
portb:=%01101101; // chiffre 5
delay_ms(1000);
portb:=%01111101; // chiffre 6
delay_ms(1000);
portb:=%00000111; // chiffre 7
delay_ms(1000);
portb:=%01111111; // chiffre 8
delay_ms(1000);
portb:=%01101111; // chiffre 9
delay_ms(1000);
end;
end.
```



Deuxième méthode :

```
program COMPTEURMOD10b;
var i : integer;
const chiffre : array[10] of byte =(%00111111, %00000110, %01011011,
%01001111, %01100110, %01101101, %01111101, %00000111,%01111111,%01101111);
begin
Trisb:=0;
```

```
portb:=0;
while (1=1) do
    begin
    for i := 0 to 9 do  // compteur
            begin
            portb:=chiffre[i]; // affichge du chiffre
            delay_ms(1000); // attente d'une seconde
            end;
    end;
end.
```

A-2 Pour pouvoir contrôler deux afficheurs sept segments avec un microcontrôleur il faut avoir au minimum 14 sorties sur ce dernier or le PIC16F84A ne possède que 13 entrée/sortie (8 sur le port B et 5 sur le port A) alors que faire ?

A-3

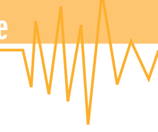
A
1
0
0 10 20 30 40 50 60 70 80 90 100 110 120 130 t(ms)

B
1
0
0 10 20 30 40 50 60 70 80 90 100 110 120 130 t(ms)

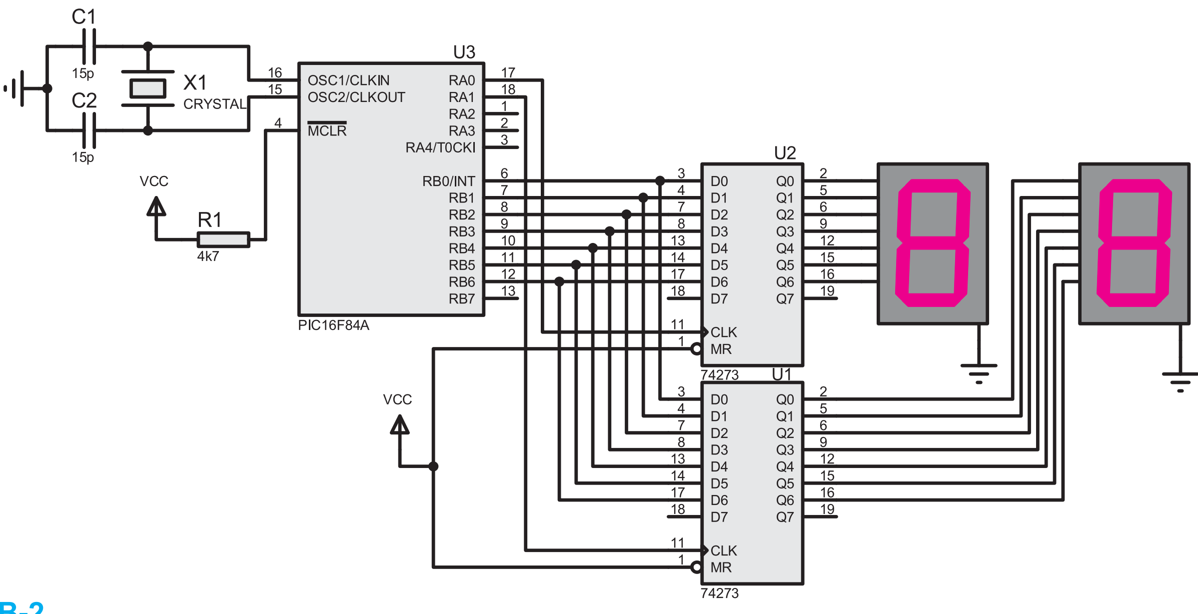
C
1
0
0 10 20 30 40 50 60 70 80 90 100 110 120 130 t(ms)

S1
1
0
0 10 20 30 40 50 60 70 80 90 100 110 120 130 t(ms)

S2
1
0
0 10 20 30 40 50 60 70 80 90 100 110 120 130 t(ms)



B-1



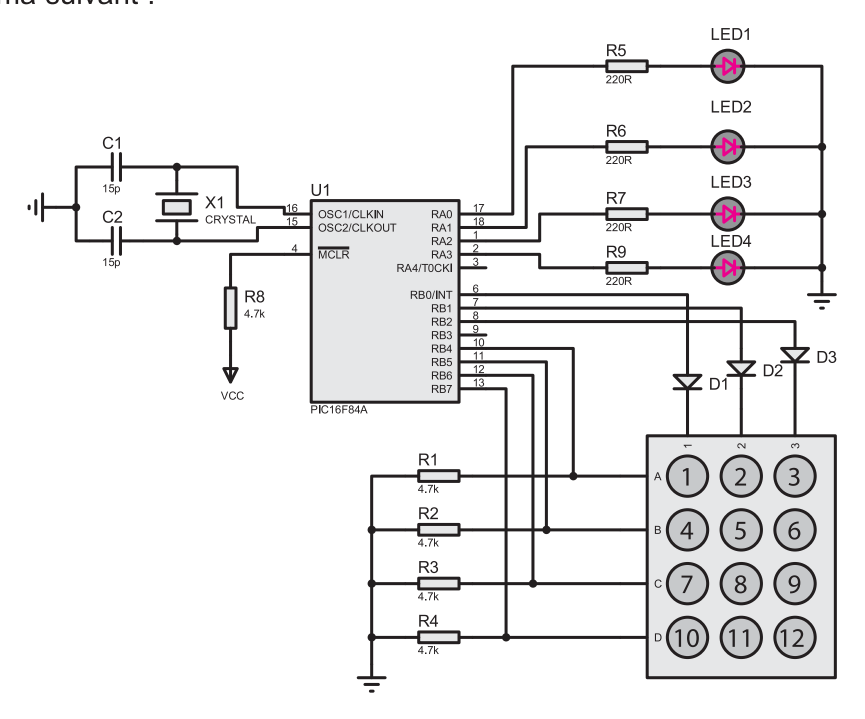
B-2

```
program COMPTEURMOD100;
var i,j : integer;
const chiffre : array[10] of byte =(%00111111, %00000110, %01011011,
%01001111, %01100110, %01101101, %01111101,
%00000111,%01111111,%01101111);
begin
Trisa:=$FC;
porta:=0;
Trisb:=0;
portb:=0;
while (1=1) do
    begin
    for j := 0 to 9 do  // compteur des dizaines
            begin
            portb:=chiffre[j]; // affichage du chiffre des dizaines
            porta.0:=1; //
            delay_ms(1); // Génération d'un front d'horloge sur RB0
            porta.0:=0;  //
            for i := 0 to 9 do  // compteur des unités
                  begin
                  portb:=chiffre[i]; // affichage du chiffre des unités
                  porta.1:=1; //
                  delay_ms(1);  // Génération d'un front d'horloge sur RB1
                  porta.1:=0; //
                  delay_ms(1000); // attente d'une seconde
                  end;
            end;
    end;
end.
```

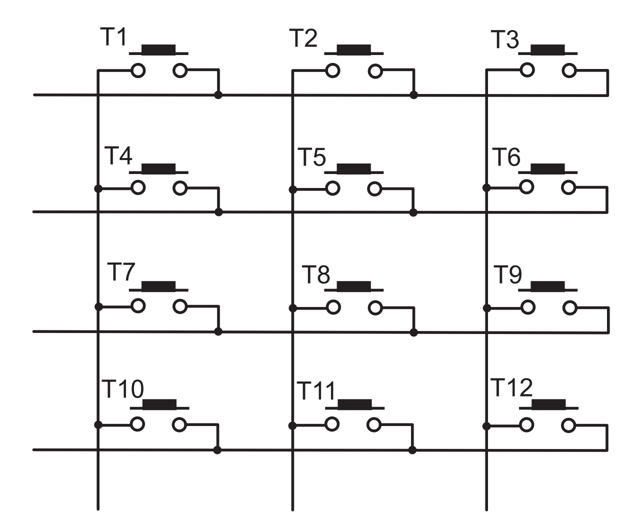


Exercice 5 : Gestion d'un clavier matriciel par le microcontrôleur PIC16F84A.

Soit le schéma suivant :



Dans un premier lieu on s'intéressera au clavier.



Le principe de ce type de clavier est simple en effet l'appui sur une touche réalise un contact direct entre la ligne et la colonne de cette touche.

Pour détecter l'action sur une touche on procède comme suit :

– On fait un balayage sur les colonnes c'est-à-dire on envoie les séquences suivantes (100, 010, 001).

– On lit l'état des lignes sur 4 bits (A, B, C, D)

On aura le tableau de correspondance suivant :

Séquence	A	B	C	D	Touche appuyée
100	0	0	0	0	Aucune touche
	1	0	0	0	T1
	0	1	0	0	T4
	0	0	1	0	T7
	0	0	0	1	T10
010	0	0	0	0	Aucune touche
	1	0	0	0	T2
	0	1	0	0	T5
	0	0	1	0	T8
	0	0	0	1	T11
001	0	0	0	0	Aucune touche
	1	0	0	0	T3
	0	1	0	0	T6
	0	0	1	0	T9
	0	0	0	1	T12

On désire afficher le code binaire correspondant à la touche appuyée sur les diodes led.

Travail demandé :

1- Traduire le fonctionnement de ce système par un algorithme.
2- Transformer cet algorithme en programme en Mikropascal.

Solution :

1- Algorithme :

```
Algorithme clavier ;
  Variables
           T : octet ;
_ Début
|     T⇐ 0 ;                                    {Initialisation de la variable T }
|     TRISA ⇐ $F0 ;                             { RA0, RA1, RA2, RA3 : sorties ;   }
|     PortA ⇐ 0 ;                               {Initialisation du portA }
|     TRISB ⇐ $F8 ;        {RB0, RB1, RB2 : sorties ; RB3, RB4, RB5, RB6, RB7 : entrées}
|     TANQUE  (1=1)  FAIRE                      {Boucle infinie}
|          _ Début
|         |   PortB  ⇐  (00000001)2 ;           {Première séquence}
|         |   Attente (10ms) ;
|         |   T⇐ PortB ;
|         |   T⇐ T décalé à droite de 4 positions ;    {Lecture de RB4, RB5, RB6 et RB7 }
|         |   SI T=8 ALORS T⇐ 10; FINSI;              {codage des touches }
|         |   SI T=4 ALORS T⇐ 7; FINSI;
|         |   SI T=2 ALORS T⇐ 4; FINSI;
|         |   SI T<> 0 ALORS PortA⇐T ; FINSI;  {Afficher le numéro de la touche appuyée}
|         |                                    {                          sur le portA   }
|         |   PortB ⇐ (00000010)2 ;                   {Deuxième séquence}
|         |   Attente (10ms) ;
|         |   T ⇐ PortB ;
|         |   T ⇐ T décalé à droite de 4 positions ;   {Lecture de RB4, RB5, RB6 et RB7}
|         |   SI T=8 ALORS T ⇐ 11 ; FINSI;             {codage des touches }
|         |   SI T=4 ALORS T ⇐ 8 ; FINSI;
|         |   SI T=2 ALORS T ⇐ 5 ; FINSI;
|         |   SI T=1 ALORS T⇐ 2 ;  FINSI;
|         |   SI T<> 0 ALORS PortA⇐T ; FINSI; {Afficher le numéro de la touche appuyée}
|         |                                   {                          sur le portA   }
|         |   PortB ⇐ (00000100)2 ;                   {Troisième séquence}
|         |   Attente (10ms) ;
|         |   T ⇐ PortB ;
|         |   T⇐T décalé à droite de 4 positions ;     {Lecture de RB4, RB5, RB6 et RB7}
|         |   SI T=8 ALORS T ⇐ 12 ; FINSI;
|         |   SI T=4 ALORS T ⇐  9  ; FINSI;
|         |   SI T=2 ALORS T ⇐  6 ; FINSI;
|         |   SI T=1 ALORS T ⇐  3 ;  FINSI;            {codage des touches }
|         |   SI T<> 0 ALORS PortA⇐T ; FINSI;{Afficher le numéro de la touche appuyée}
|         |                                  {                          sur le portA   }
|         |_FinFaire ;
|_Fin.
```





2- Programme

```
program clavier;
  Var T : byte ;
begin
T:= 0 ;                          {Initialisation de la variable T  }
TRISA := $F0 ;                   { RA0, RA1, RA2, RA3 : sorties ;   }
portA:=0;                        {Initialisation du portA }
TRISB :=$F8 ;    {RB0, RB1, RB2 : sorties ; RB3, RB4, RB5, RB6, RB7 : entrées }
while  (1=1)  do                 {Boucle infinie}
    begin
          PortB  := %00000001 ;                {Première séquence}
          delay_ms(10) ;
         T:= PortB ;
         T:= T shr 4  ;                        {Lecture de RB4, RB5, RB6 et RB7}
         if T=8 then T:= 10;                   {codage des touches }
         if T=4 then T:= 7;
         if T=2 then T:= 4;
         if T<> 0 then PortA:=T ;    {Afficher le numéro de la touche appuyée sur le portA  }
         PortB :=%00000010 ;                   {Deuxième séquence}
        delay_ms(10) ;
        T := PortB ;
        T := T shr 4  ;                        {Lecture de RB4, RB5, RB6 et RB7}
        if T=8 then T := 11 ;                  {codage des touches }
        if T=4 then T := 8 ;
        if T=2 then T := 5 ;
        if T=1 then T := 2 ;
        if T<> 0 then PortA := T ;   {Afficher le numéro de la touche appuyée sur le portA   }
        PortB := %00000100 ;                   {Troisième séquence}
       delay_ms(10) ;
       T := PortB ;
       T := T shr 4  ;                         {Lecture de RB4, RB5, RB6 et RB7}
       if T=8 then T := 12 ;
       if T=4 then T :=  9 ;
       if T=2 then T :=  6 ;
       if T=1 then T :=  3 ;                   {codage des touches }
       if T<> 0 then PortA := T ;    {Afficher le numéro de la touche appuyée sur le portA  }
    end ;
end.
```

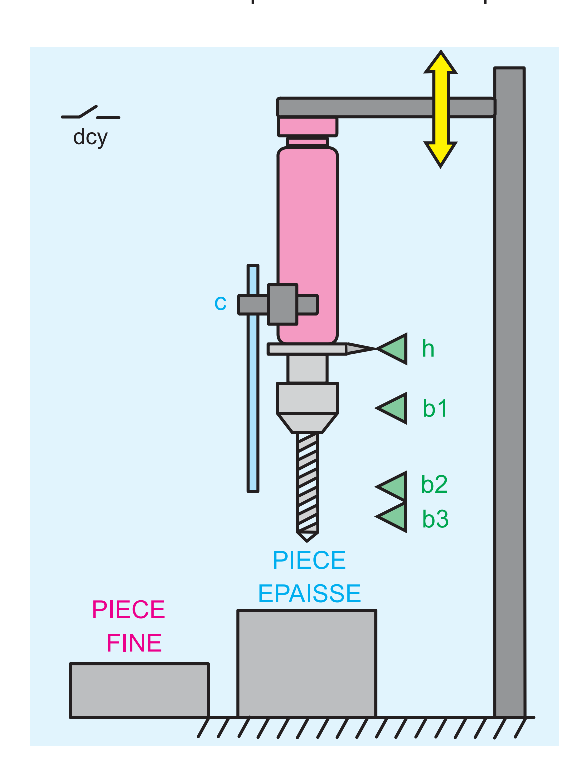
2- Exercices à résoudre :

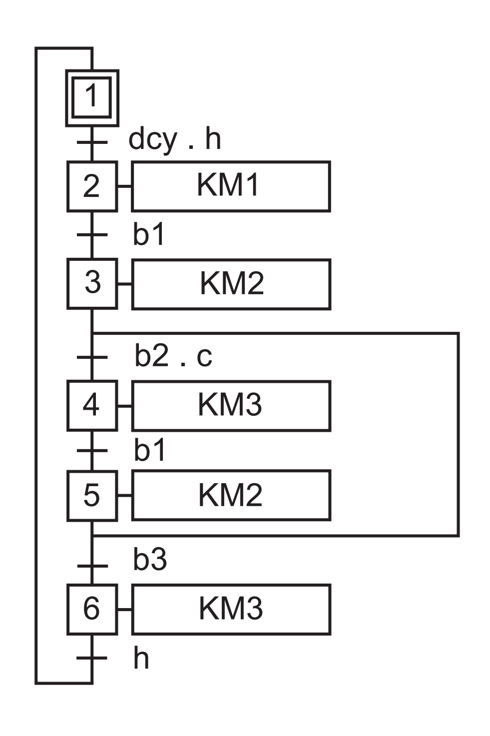
Exercice 1 : (Perceuse automatique)

Le système présenté est une perceuse qui est insérée dans une ligne de production continue. Un opérateur place la pièce devant être percée et la serre dans un étau.

Suite à l'action sur le bouton départ cycle"dcy" la perceuse effectue, selon l'épaisseur de la pièce, un cycle avec ou sans débourrage. On vous donne le Grafcet d'un point de vue de la partie commande.

KM1 : contacteur pour la descente rapide.
KM2 : contacteur pour la descente lente.
KM3 : contacteur pour la montée rapide.



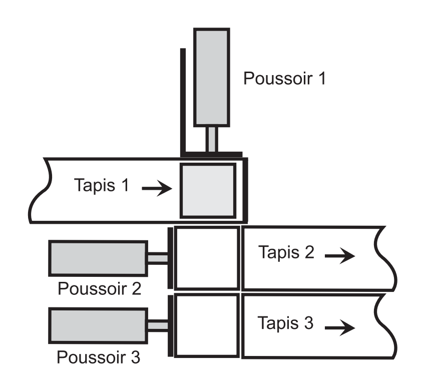


Travail demandé :

1- Représenter le fonctionnement par un algorithme.
2- Traduire cet algorithme en un programme en mikropascal.

Exercice 2 : (trieuse de caisses)

Un dispositif automatique destiné à trier des caisses de deux tailles différentes se compose d'un tapis, amenant les caisses, de trois poussoirs et deux tapis d'évacuation, comme l'indique la figure ci-contre.



Vue de dessus du dispositif de tri





Fonctionnement :
Le poussoir 1 pousse les petites caisses devant le poussoir 2 qui les transfère à son tour sur le tapis d'évacuation 2, alors que les grandes caisses sont poussées devant le poussoir 3, ce dernier les évacue sur le tapis d'évacuation 3.
Pour effectuer la sélection des caisses, un dispositif de détection placé devant le poussoir 1, permet de reconnaître le type de caisse qui se présente.

Travail demandé :

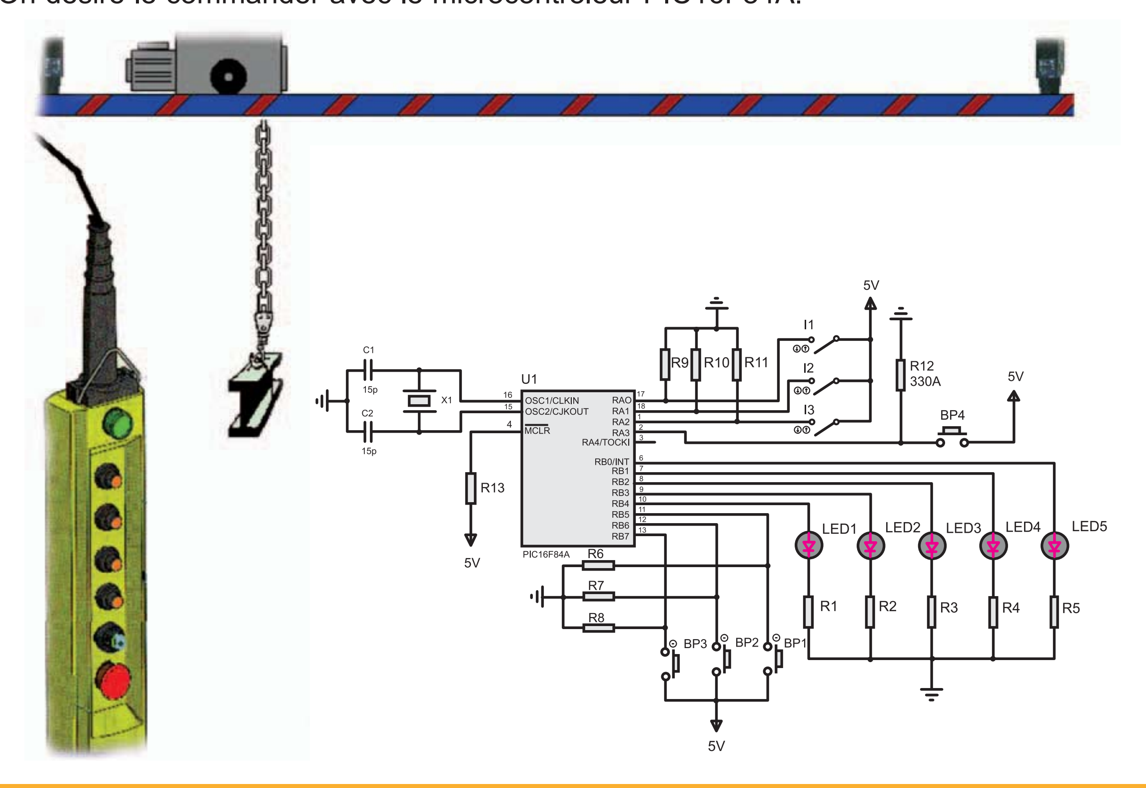
1- Etablir la liste des capteurs et des actionneurs nécessaires à ce système.
2- Remplir le tableau suivant :

Actionneur	Pré Act.	Broche du microcontrôleur

Capteur	Broche du microcontrôleur

3- Représenter par un algorithme le fonctionnement de la commande de ce dispositif.
4- Traduire cet algorithme en un programme en mikropascal.

Exercice 3 : (pont roulant)
Ce système est généralement installé dans les usines ou dans les dépôts de marchandises, il nous permet de déplacer des charges de poids important.
On désire le commander avec le microcontrôleur PIC16F84A.



La configuration des entrées/sorties est donnée dans le tableau suivant :

Interrupteur 'marche'	RA0	I1
Bouton poussoir 'descente'	RB5	BP1
Bouton poussoir 'gauche'	RB6	BP2
Bouton poussoir 'droite'	RB7	BP3
Bouton poussoir 'montée'	RA3	BP4
Butée gauche du chariot	RA1	I2
Butée droite du chariot	RA2	I3
Lampe 'Présence tension'	RB0	LED5
Sortie vers relais 'gauche'	RB1	LED4
Sortie vers relais 'droite'	RB2	LED3
Sortie vers relais 'montée'	RB3	LED2
Sortie vers relais 'descente'	RB4	LED1

Le bouton d'arrêt d'urgence coupe l'alimentation sur le système.
L'appui sur deux boutons en même temps pour un déplacement en sens contraires provoque l'arrêt.

On vous donne les équations des sorties :

$$LED = I1.BP1.\overline{BP4}$$

$$LED2 = I1.BP4.\overline{BP1}$$

$$LED3 = I1.BP3.\overline{BP2}.\overline{I3}$$

$$LED4 = I1.BP2.\overline{BP3}.\overline{I2}$$

$$LED5 = 1$$

Travail demandé :

1- Etablir l'algorithme correspondant.
2- Traduire le fonctionnement par un programme en mikropascal.

ASSERVISSEMENT LINÉAIRE

A- MISE EN SITUATION

1- Présentation du système :

Système de positionnement d'une antenne parabolique



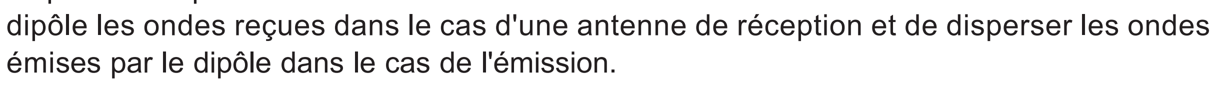
Un faisceau hertzien est une liaison haute fréquence "point à point " destinée à véhiculer sur une fréquence porteuse un signal analogique ou un signal numérique.

NB :
Hautes fréquences : ce sont les fréquences allant approximativement de 300 kHz à 30GHz.
Point à point : signifie que le tir du faisceau est dirigé de l'émetteur (fixe ou mobile) vers un récepteur bien ciblé, en général fixe, contrairement au broadcasting qui "arrose" l'espace pour atteindre une infinité de récepteurs (exemple : émetteur radio, TNT, WiFi)

Une antenne parabolique pour faisceau hertzien est un ensemble constitué d'un dipôle rayonnant et d'un réflecteur parabolique ...

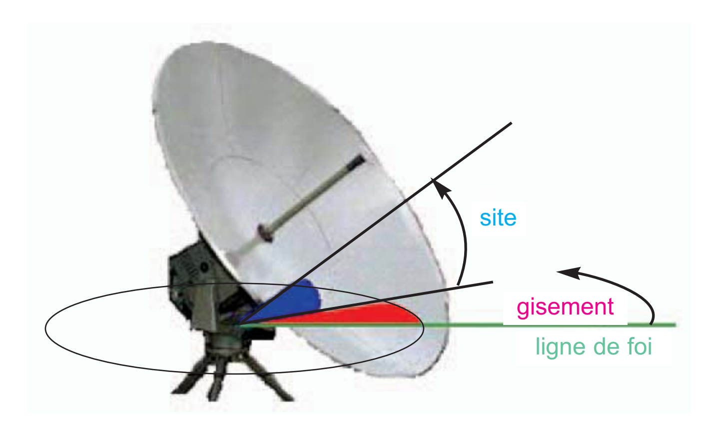
Le dipôle est fixé au bout du bracon central : c'est la partie sensible de l'antenne, capable d'émettre et/ou de recevoir des signaux électriques. Il occupe le foyer de la parabole.

La parabole a pour fonction de rassembler sur ce dipôle les ondes reçues dans le cas d'une antenne de réception et de disperser les ondes émises par le dipôle dans le cas de l'émission.



Cette antenne étant destinée à communiquer avec un autre émetteur-récepteur. Elle doit être orientable dans l'espace. Elle est donc **réglable** autour de deux axes afin de diriger l'antenne dans la direction de **réception maximale**.

La présence de la surface de réflexion parabolique et l'existence de son lobe principal font que cette puissance va se trouver "**concentrée**" dans le lobe.



Cette direction est repérée par deux angles :

- le **gisement**, angle horizontal gradué de 0 à 360° à partir d'une direction dite ligne de foi
- le **site** angle vertical gradué de 0 à 90°...

La connaissance et le **réglage** de ces **deux angles** assurent le pointage de l'antenne vers sa **cible** (une autre antenne, un satellite, un avion...)

Gisement et **site** sont des termes empruntés à la terminologie militaire (artillerie) ou aérienne et navale (radar de poursuite)... ils ont pour correspondants "civils" **azimut** et **hauteur.**

2- Problèmes posés :

a) Que peut-on utiliser pour avoir la réception maximale ?
b) Quel procédé peut-on utiliser pour positionner automatiquement une parabole ?
c) Comment atteindre rapidement la bonne position et avec la meilleure précision ?

B- RAPPEL

Activité

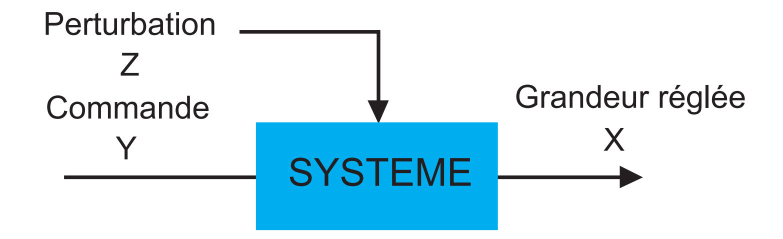
Réaliser l'activité N°1 du TP-C1 dans le manuel d'activités

Il s'agit, de commander un moteur à courant continu :

- en boucle ouverte
- en boucle fermée

1- Fonctionnement en boucle ouverte ou en mode manuel

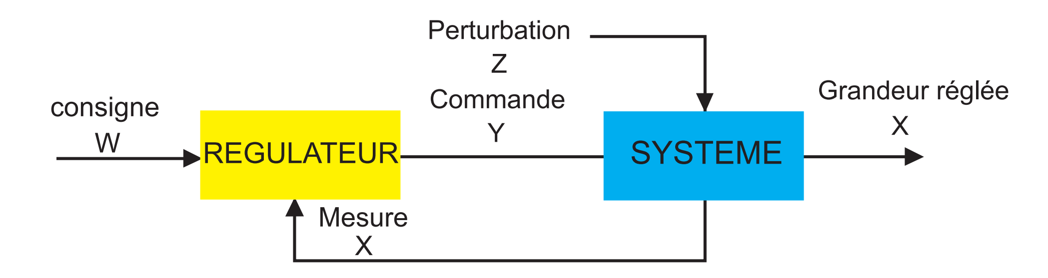
On parle de fonctionnement en **boucle ouverte** quand c'est l'**opérateur** qui contrôle l'**organe de réglage**. Ce n'est pas un asservissement.





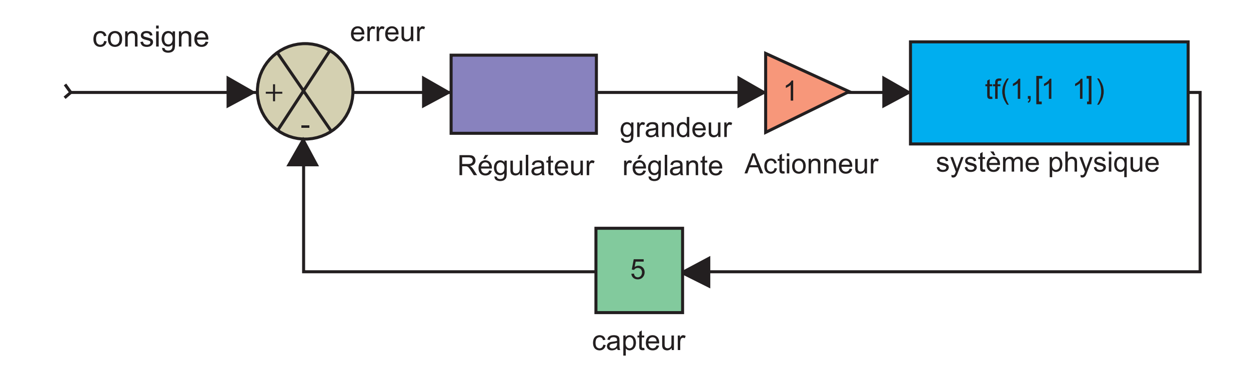
2- Fonctionnement en boucle fermée ou en mode automatique

C'est le fonctionnement normal d'un système asservi. Le processus compare la mesure de la grandeur réglée à la consigne désirée et agit en conséquence pour s'en rapprocher.



3- Structure générale d'un système asservi

Les asservissements, qui sont des systèmes de commande en boucle fermée, sont constitués, dans la plupart des cas, de la façon suivante :
Les principaux organes en sont :



- **le système physique :**
  il génère la variable que l'on désire asservir,
- **l'actionneur ou organe de puissance :**
  il peut être inclus dans le système physique à asservir,
- **le capteur :**
  il réalise la mesure de la grandeur commandée,
- **le comparateur :**
  il calcule la différence entre la grandeur désirée et la grandeur obtenue (c'est-à-dire l'erreur),
- **le régulateur :**
  c'est l'organe de commande : son rôle consiste à ajuster l'action à partir de l'erreur, Il élabore la variable qui va agir et commander l'actionneur
- **les perturbations :**
  ce sont des modifications non prévisibles sur le système.



4- Régulation et asservissement :

On appelle **régulation** un système asservi qui doit maintenir **constante la sortie** conformément à une **consigne constante**, indépendamment des perturbations.
Ex : Régulation de température.

On appelle **asservissement** un système asservi dont la **sortie doit suivre** le plus fidèlement possible une **consigne variable**.
Ex : suivi de trajectoire.

5- Schéma fonctionnel d'un système asservi

Le schéma fonctionnel tente de représenter les relations entre les différentes grandeurs physiques des boucles de régulation. Il sera composé uniquement des éléments suivants :

- **Ligne de parcours d'une grandeur physique** :
Cette ligne représente le parcours d'une même grandeur physique de la boucle de régulation.

Grandeur physique

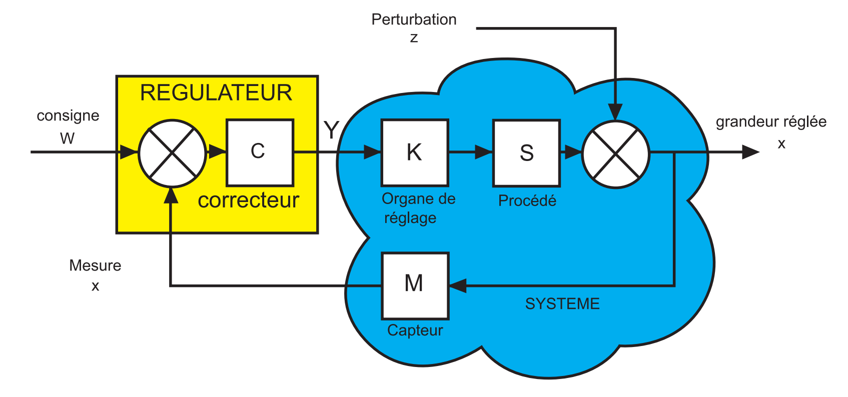
- **Bloc gain :**
Le bloc représente la relation entre deux grandeurs physiques, relation réaliser par un élément de la boucle de régulation :



- **Sommateur et soustracteur :**
Ce bloc représente l'addition ou la soustraction de grandeurs physique de même nature.



- **Représentation fonctionnelle d'une boucle de régulation**
Elle regroupe les diiférents blocs qui constituent les deux chaines d'action et de réaction (chaîne d'informations)





C- PERFORMANCES D'UN SYSTEME ASSERVI

Activité

Réaliser l'activité N°2 du TP-A50 dans le manuel d'activités

Il s'agit d'utiliser le logiciel " Correcteur PID " Pour mettre en évidence le comportement d'une boucle d'asservissement (brancher et débrancher la boucle). Modifier les actions du correcteur et analyser les performances vis à vis d'une perturbation.

1- Critères de performance d'un système asservi :

Le choix du correcteur va dépendre des performances attendues du système en boucle fermée. Ces critères sont :

- **La stabilité :** le système est stable si sa sortie évolue indéfiniment sans modification de l'entrée (régime transitoire infini),
- **La rapidité** : c'est le temps que met le système à réagir pour arriver à sa valeur finale,
- **La précision** : elle est définie par l'erreur entre la valeur finale souhaitée et celle qui est réellement atteinte.

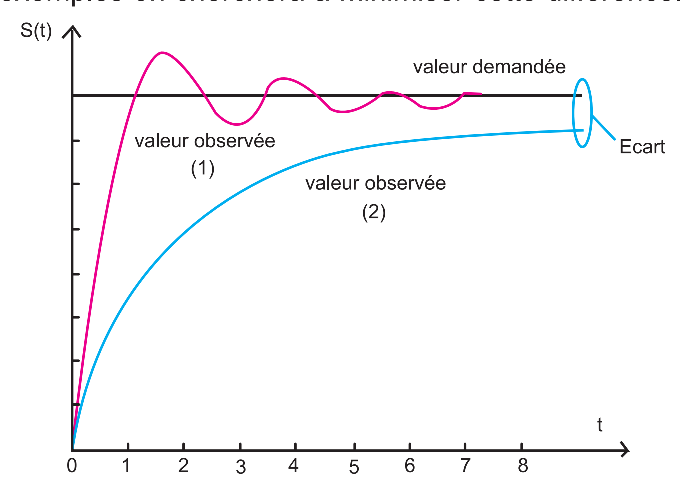
Il est alors nécessaire de développer plusieurs types de correcteurs agissant sur les différentes défaillances qui peuvent perturber **la qualité** de fonctionnement d'un système asservi. Trois actions sont possibles :

- **L'action proportionnelle P** : elle agit de manière proportionnelle à l'erreur (par exemple, si l'erreur est nulle, cette action est nulle),
- **L'action intégrale I** : elle agit sur l'intégrale de l'erreur,
- **L'action dérivée D** : elle agit de manière proportionnelle à la dérivée de l'erreur.

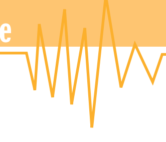
2- Critère de précision d'un système asservi :

Pour une consigne (valeur demandée ou souhaitée), la sortie observée se stabilise à une valeur finale (régime permanent, valeur finale observée). **La précision** est la différence entre ces deux valeurs exprimées en %.

Dans de nombreux exemples on cherchera à minimiser cette différence.



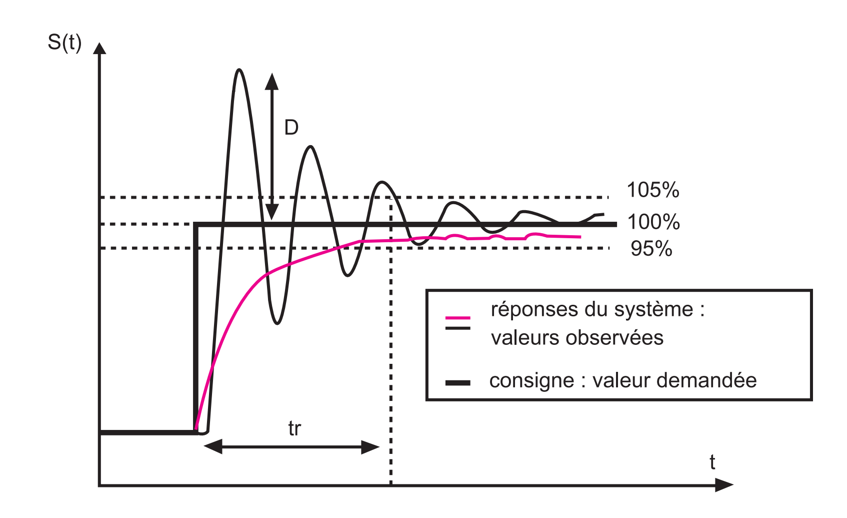
NB : Si la sortie consignée est une constante, l'écart est appelé : "**écart statique**"



3- Critère de rapidité d'un système asservi :

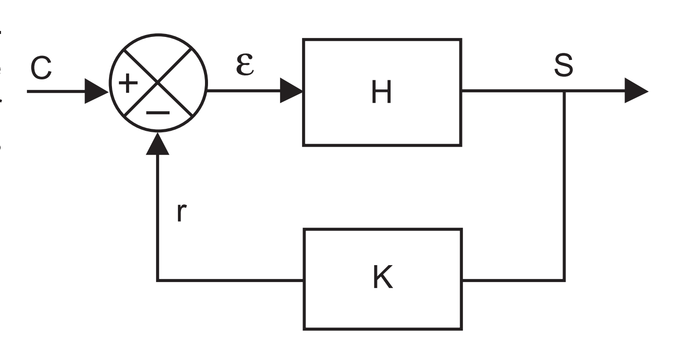
On note **t_r** le temps de réponse à 5%, ou encore $t_{5\%}$ l'instant à partir duquel l'écart par rapport à la valeur finale observée reste inférieur à 5% (de cette valeur finale) :

- Si $t_{5\%}$ est "grand", le système est dit lent
- Si $t_{5\%}$ est "petit", le système est dit rapide
- D est le dépassement.



4- Critère de stabilité d'un système asservi :

Un système asservi est instable s'il fonctionne en oscillations entretenues. Pour le rendre stable, on est amené à modifier ses paramètres de réglage pour ne pas avoir une transmitrance **T** infinie :



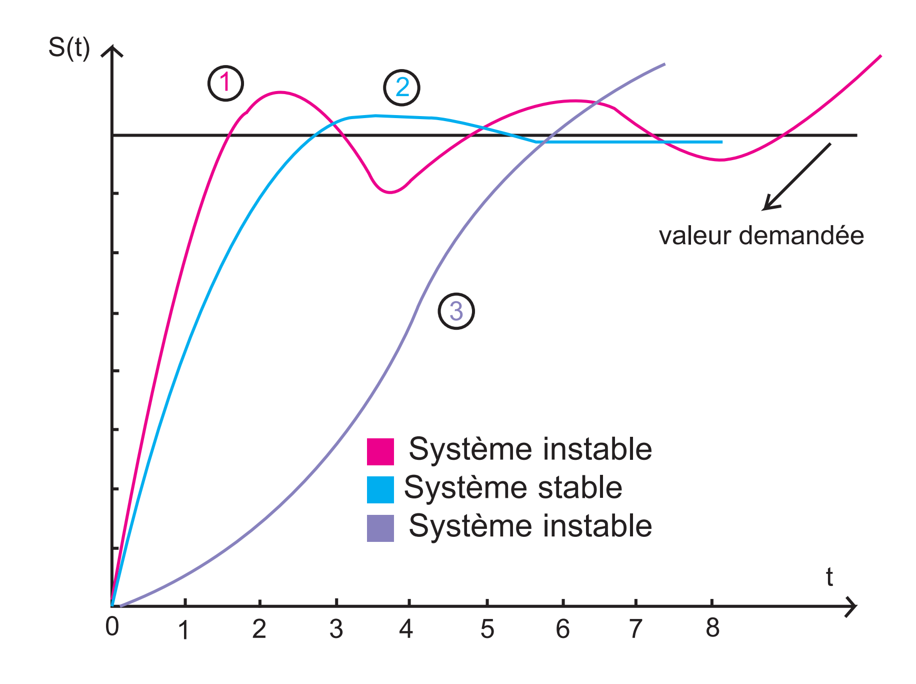
$S = H.\varepsilon$, $\quad .\varepsilon = C - r$,
$r = K.S$, $\quad S = H.(C - K.S)$

d'où : $T = \frac{S}{c} = \frac{H}{1 + KH}$

Pour que la transmittance ne soit pas infinie, le dénominateur **1+ KH** doit être différent de zéro. Dans la majorité des applications il est nécessaire de corriger **H** en ajoutant un organe de réglage dans la chaine d'action. L'organe à insérer porte le nom de correcteur du système asservi.

L'objectif d'un asservissement est que la sortie puisse suivre dans la mesure du possible une entrée de référence. Si la sortie converge vers la consigne le système est considéré comme stable, dans le cas contraire il est instable.

Le tracé des réponses instantanées suivantes montre le comportement de trois systèmes différents :



E- CORRECTION DES SYSTEMES ASSERVIS

Activité

Réaliser l'activité N°3 du TP-A5 dans le manuel d'activités

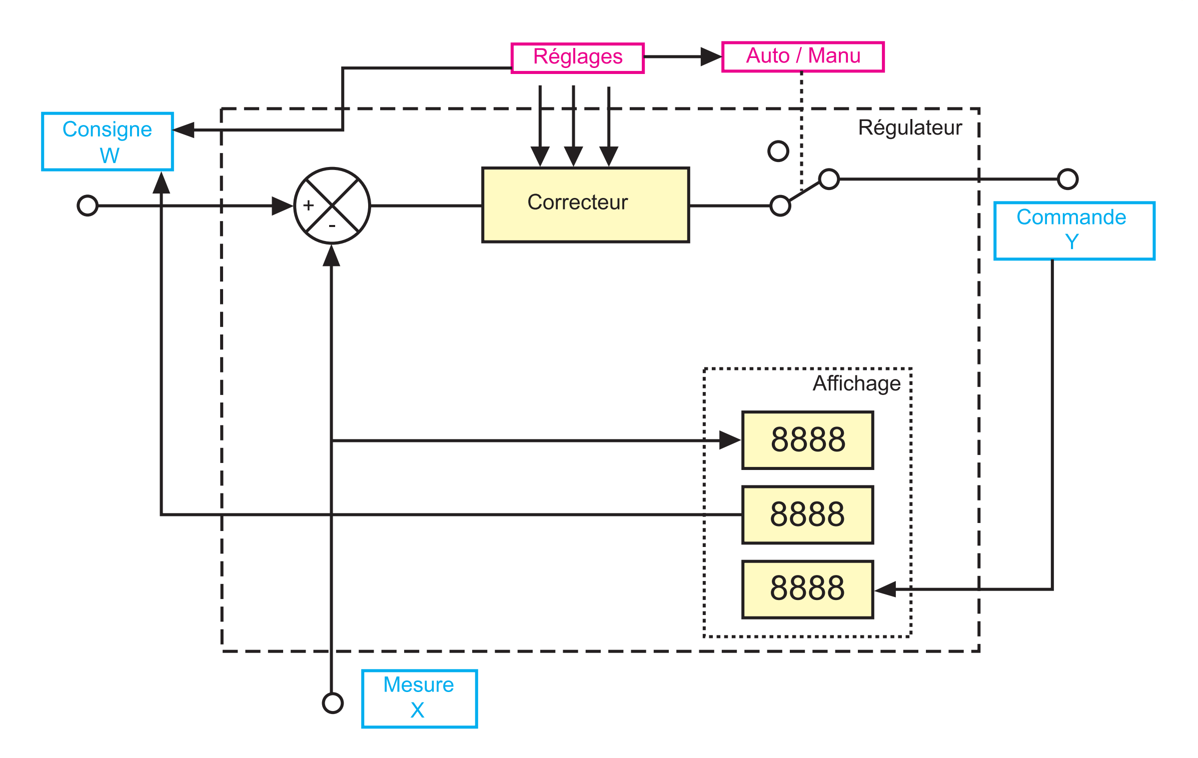
Il s'agit d'utiliser le logiciel " Correcteur PID " pour :

- mettre en évidence le comportement d'une boucle d'asservissement (brancher et débrancher la boucle)
- modifier les actions du correcteur et analyser les performances vis à vis d'une perturbation.

1- Nécessité des correcteurs

Dans tout système asservi la grandeur de sortie doit recopier le mieux possible la consigne. Il n'y a malheureusement pas de paramètres qui permettent de rendre un système à la fois rapide, précis et stable. On remarque notamment qu'une augmentation du gain en boucle ouverte améliore la rapidité et la précision mais rend le système plus instable, en boucle fermée.

L'automaticien est souvent confronté à ce dilemme **stabilité-précision** et ne dispose que d'une marge de manoeuvre réduite. Les organes de puissance, et souvent les capteurs, sont imposés. C'est la raison pour laquelle l'**insertion** d'**un** ou **plusieurs correcteurs** devient obligatoire.

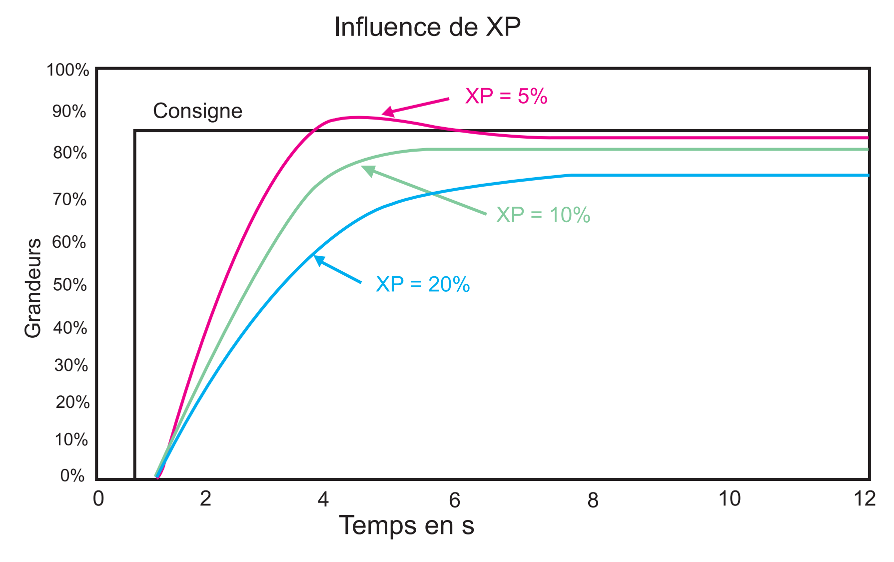


2- Action proportionnelle

Elle permet de doser la commande en fonction de l'éloignement du but cherché. Plus l'écart est important plus la réaction doit être vive.

a/ Comportement dynamique (fonctionnement en régime transitoire) :

Plus la bande proportionnelle est petite, plus le temps de réponse du système est court. En effet, pour la même erreur, la commande fournie est plus importante qu'avec une bande proportionnelle plus importante. Si la **bande proportionnelle** se **rapproche de 0**, le système devient **instable**. En effet, un fonctionnement en TOR correspond à une bande proportionnelle nulle.





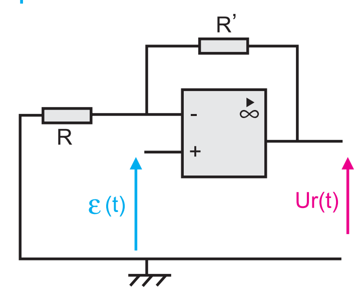
b/ Matérialisation d'une action proportionnelle

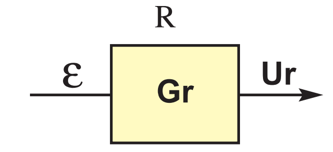
$Ur = (R + R') . i$

$\varepsilon = R . i$ donc $i = \varepsilon / R$

d'où $Ur = (R + R') . \varepsilon / R$

alors : $Ur(t) = (1 + \frac{R'}{R})\, \varepsilon(t)$





- Si G_r est grand, la correction est énergique mais n'est pas sans danger (pompage).
- Si G_r est faible, la correction est molle et lente, mais il n'y a pas de risque.

NB : l'erreur statique

si $\varepsilon = 0$, alors Ur = 0, le processus n'est plus alimenté ! La loi de commande proportionnelle doit donc s'écrire : $ur = Gr\,\varepsilon + Uro$ où Uro est une tension résiduelle nécessaire, mais qui va provoquer une erreur statique. l'erreur statique diminue lorsqu'on augmente le gain, mais elle peut être importante si celui-ci est faible.

2- Action Intégrale

L'action proportionnelle se montre donc insuffisante pour régler, seule, les imperfections d'un système, en particulier, lorsqu'on désire obtenir une précision inférieure à l'erreur statique. Comme pour la conduite d'une voiture, au lieu d'écraser l'accélérateur, on peut avoir avantage à assurer une commande plus progressive. Cette commande est obtenue par une loi intégrale : Ti est la constante d'intégration : c'est le temps au bout duquel la sortie a répété l'entrée.

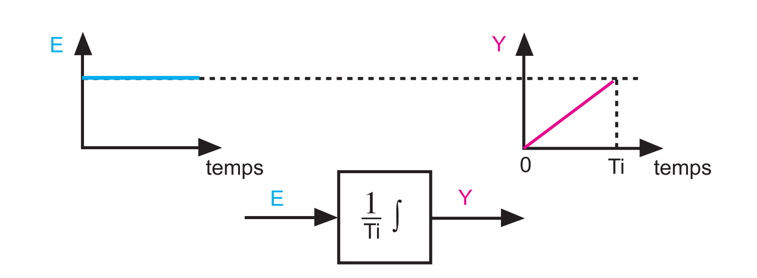
a/ Fonctionnement

Pour étudier l'influence de l'action intégrale, on s'intéressera à la réponse du module intégral à un échelon. Plus **Ki** est grand (Ti petit), plus la valeur de la sortie Y augmente rapidement. Le temps Ti est le temps pour que la commande Y augmente de la valeur de l'entrée **E = W - X**.

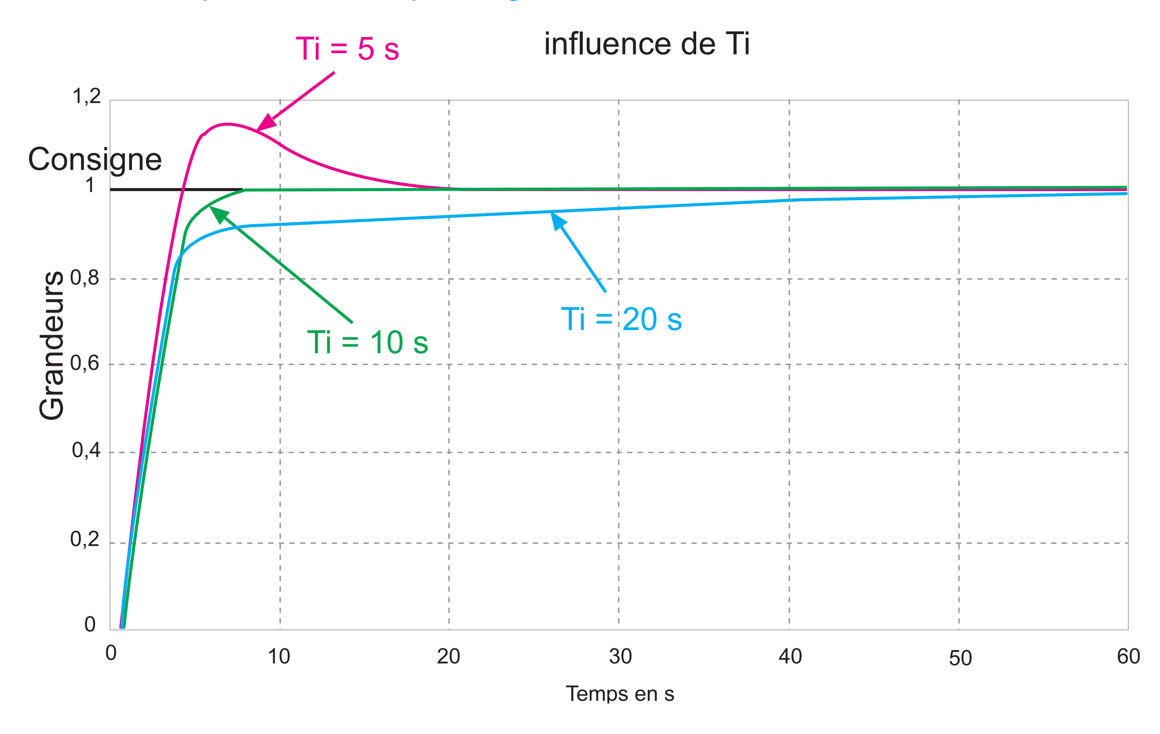
$$U_r(t) = \frac{1}{T_1}\int_0^t \varepsilon(x)\,dx$$

$$U_r = \frac{E}{T_1}.t$$

donc $u_r = E$ et $t = t_o + T_i$



b/ Influence du paramètre temps intégral

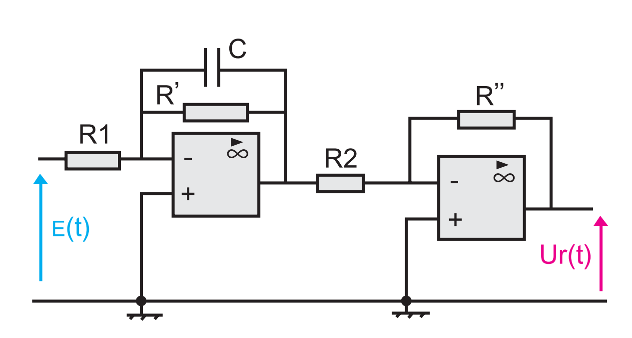


Comportement statique : Quelle que soit la valeur de l'action intégrale, l'erreur statique est nulle (si le système est stable).

Comportement dynamique : Lors d'une réponse indicielle, plus **Ti** est petit plus le système se rapproche de l'instabilité.

c/ Matérialisation d'une action intégrale

La loi de commande intégrale est donc progressive. On dit encore qu'elle est **persévérante**. Tant que l'erreur statique ($\varepsilon > 0$ ou $\varepsilon < 0$) existe, l'action intégrale agit (positivement ou négativement) jusqu'à ce que celle-ci s'annule. Toujours par analogie avec la conduite d'une voiture, rien ne sert de conduire par à-coups : il vaut mieux appuyer progressivement sur l'accélérateur, laisser la voiture atteindre la vitesse désirée, puis conserver celle-ci en maintenant le pied à la même hauteur.



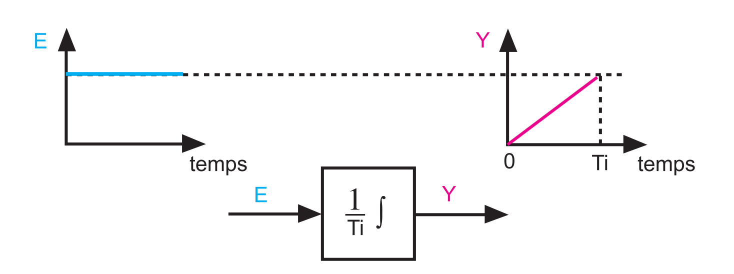
3- Action dérivée

C'est une action qui amplifie les variations brusques de la consigne. Elle a une action opposée à l'action intégrale. Cette fonction est remplie par l'opérateur mathématique “dériver par rapport au temps”. Ainsi, dans un régulateur, on définit l'action “dérivé” à partir du temps dérivé Td.

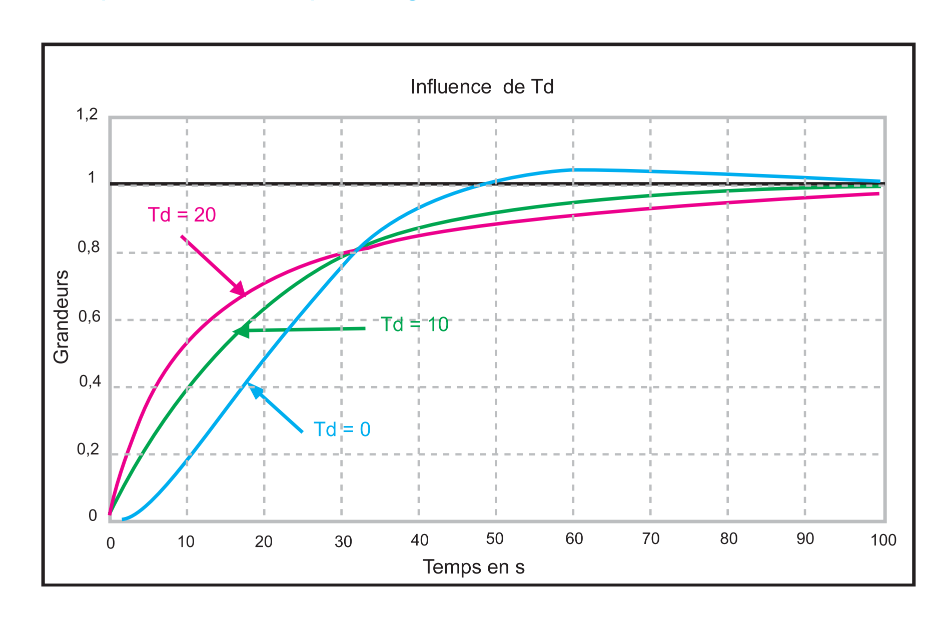


a/ Fonctionnement

Pour étudier l'influence de l'action dérivée, on s'intéressera à la réponse du module dérivé à une rampe. Plus Td est grand, plus la valeur de la sortie Y sera importante. Le temps Td est le temps pour que l'entrée E augmente de la valeur de la sortie Y.



b/ Influence du paramètre temps intégral



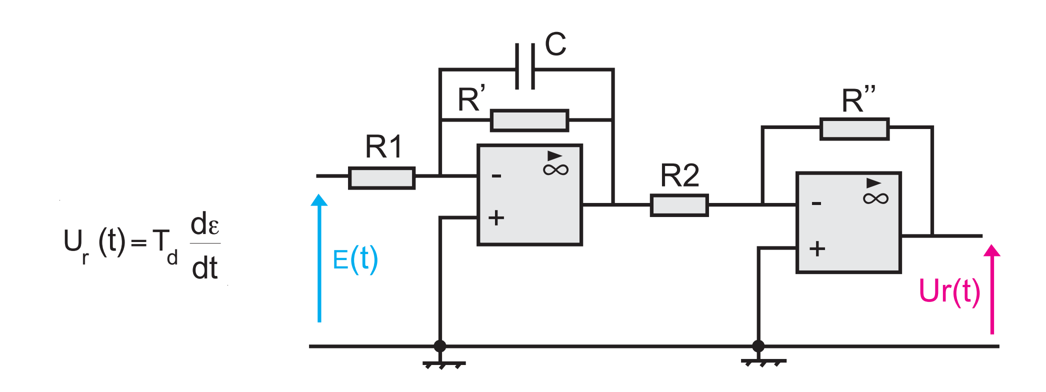
Comportement statique : Aucune influence.

Comportement dynamique : Lors d'une réponse indicielle, plus Td est grand plus le système est rapide.

c/ Matérialisation d'une action dérivée

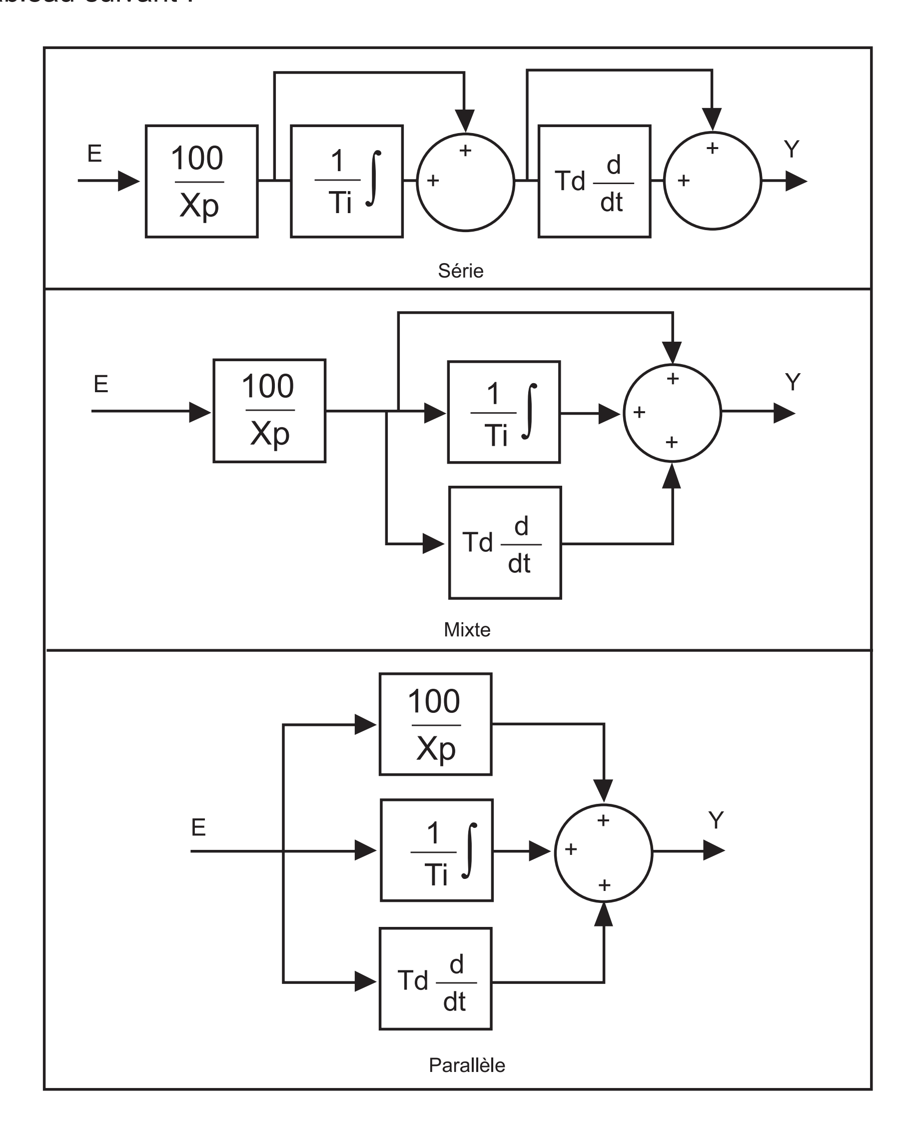
L'action dérivée améliore la rapidité de la chaîne asservie.

$$U_r(t) = T_d \frac{d\varepsilon}{dt}$$



4- Action conjuguée

En général, le régulateur ne fonctionne pas en action dérivée pure (trop instable). Il fonctionne en correcteur Proportionnel Intégral Dérivé (PID). Le triplet, Bande Proportionnelle - Temps Intégral - Temps dérivé, définit trois types de fonctionnement qui sont représentés dans le tableau suivant :



Remarque : Les régulateurs électroniques (tous les systèmes industriels) ont une structure mixte.

5- Résumé des actions des corrections P, I et D

Quand ... augmente	Stabilité	Rapidité	Précision
Bande proportionnelle = Xp	Augmente	Diminue	Diminue
Temps intégral = Ti	Augmente	Diminue	Pas d'influence
Temps dérivé = Td	Diminue	Augmente	Pas d'influence

F- RESUMÉ

L'action Proportionnelle

L'action Proportionnelle corrige de manière instantanée, donc rapide, tout écart de la grandeur à régler, elle permet de vaincre les grandes inerties du système. afin de diminuer l'écart de réglage et rendre le système plus rapide, on augmente le gain (on diminue la bande proportionnelle) mais, on est limité par la stabilité du système.

Le correcteur P est utilisé lorsqu'on d"sire régler un paramètre dont la précision n'est pas importante, exemple : régler le niveau d'eau dans un bac de stockage.

L'action Intégrale

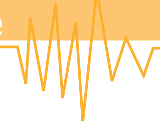
L'action intégrale complète l'action proportionnelle. Elle permet d'éliminer l'erreur résiduelle en régime permanent. Afin de rendre le système plus dynamique (diminuer le temps de réponse), on diminue l'action intégrale mais, ceci provoque l'augmentation du déphasage ce qui provoque l'instabilité en boucle fermée.

L'action intégrale est utilisée lorsqu'on désire avoir en régime permanent, une précision parfaite, en outre, elle permet de filtrer la variable à régler d'où l'utilité pour le réglage des variables bruitées telles que la pression.

L'action Dérivée

L'action Dérivée, en compensant les inerties dues au temps mort, accélère la réponse du système et améliore la stabilité de la boucle, en permettant notamment un amortissement rapide des oscillations dues à l'apparition d'une perturbation ou à une variation brusque de la consigne. Dans la pratique, l'action dérivée est appliquée aux variations de la grandeur à régler seule et non de l'écart mesure-consigne afin d'éviter les à coups dus à une variation subite de la consigne.

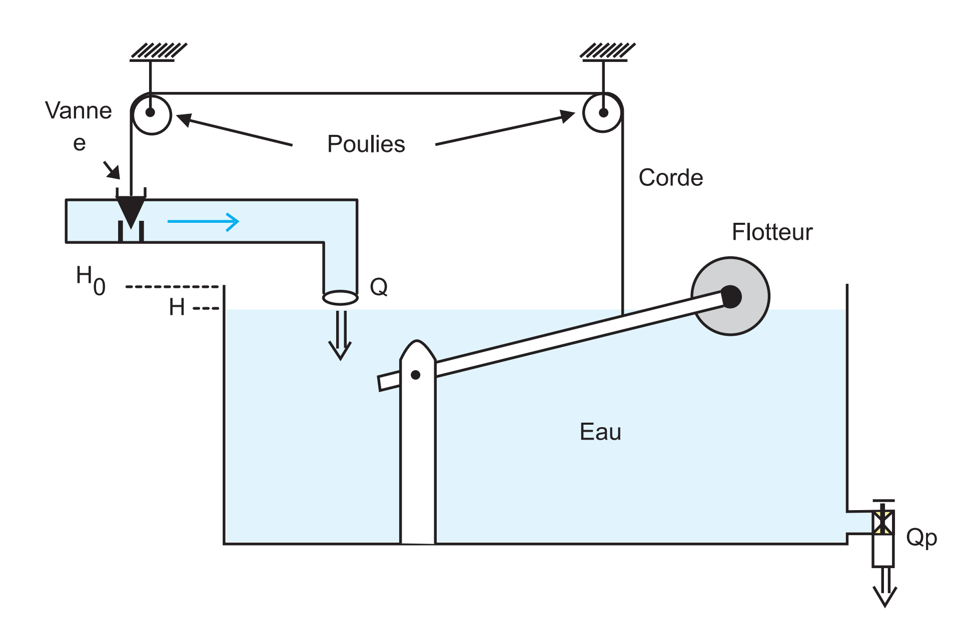
L'action D est utilisée dans l'industrie pour le réglage des variables lentes telles que la température, elle n'est pas recommandée pour le réglage d'une variable bruitée ou trop dynamique (la pression). En dérivant un bruit, son amplitude risque de devenir plus importante que celle du signal utile.



G- EXERCICES D'APPLICATION

Exercice résolu : Régulation de niveau

Tracer le schéma fonctionnel d'une chasse d'eau de la figure suivante :

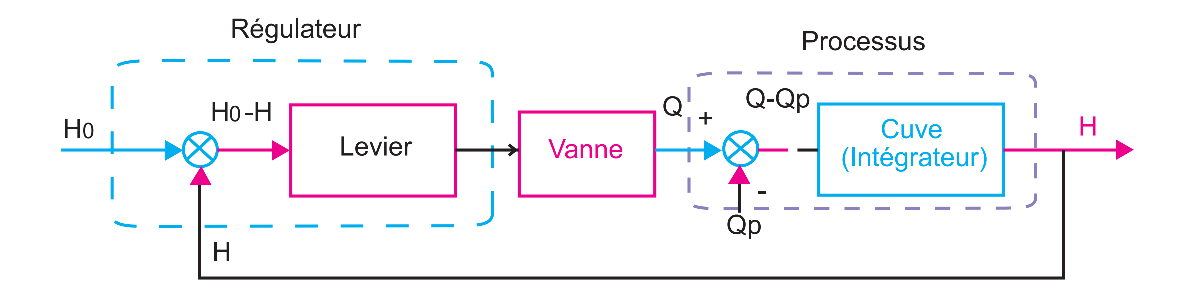


La vanne provoque l'écoulement d'un débit **Q** qui dépend du déplacement (**e**), lui-même fonction de la différence de hauteur d'eau (**Ho-H**) et des dimensions du levier et de la corde supportant le flotteur. Quand **H** est égal à **Ho**, le débit **Q** doit être nul.

La difficulté est de déterminer l'entrée et la sortie. l'erreur à ne pas commettre est de considérer l'entrée **Q** et la sortie **Qp**. L'entrée est la valeur souhaitée, la sortie est la valeur observée, soit respectivement **Ho** et **H**.

La cuve est un intégrateur car $H = \int \frac{Q - Q_p}{S}.dt$ Sachant que : $Q = S.\frac{dH}{dt}$

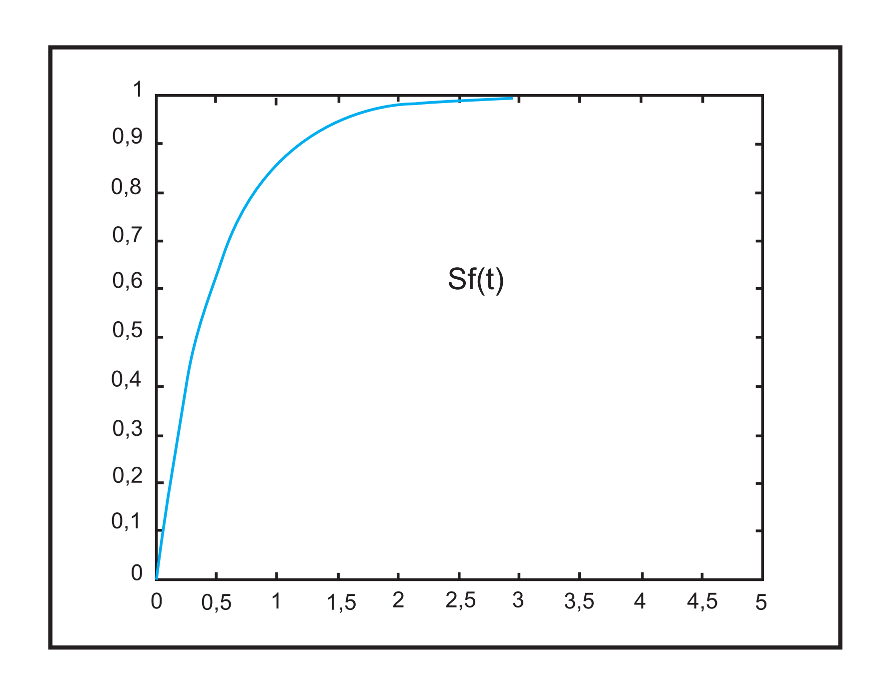
Solution :



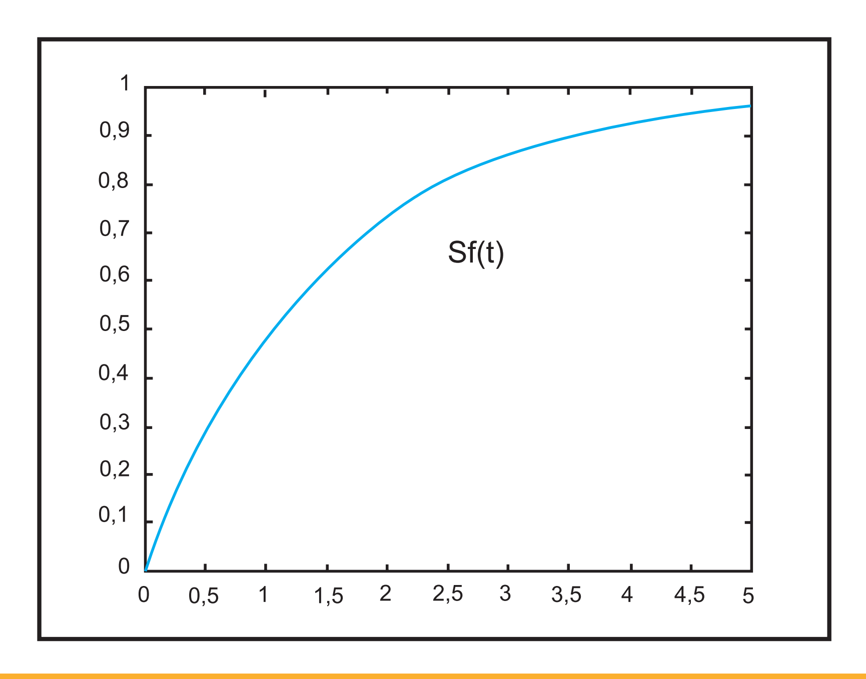


Exercices à résoudre :

Exercice 1 :
Déterminer le temps de réponse à 5% de l'allure instantannée de la sortie d'un système du premier ordre.



Exercice 2 : Même question que l'exercice N°1.



Electrotechnique

B1 Circuits électriques en courant alternatif

B2 Transformateur monophasé

B3 Moteur asynchrone

B4 Moteur à courant continu

Partie Electrotechnique

CIRCUITS EN MONOPHASÉ

A- MISE EN SITUATION

1- Fonctionnement du système : Télécommande centralisée.

Régulation de la production d'énergie électrique

En Tunisie, nous utilisons principalement deux sources de production d'énergie électrique :
- Production par les centrales thermiques.
- Production par les centrales hydrauliques.

NB : la production par les centrales nucléaires est prévue pour l'année 2020.

Pour répondre aux critères de rentabilité, la production d'énergie doit être liée à sa consommation, qui varie en fonction des saisons et des heures de la journée.
Pour cela le réseau tunisien de production d'énergie électrique est interconnecté avec les pays voisins, "un projet de liaison avec l'Europe à travers le réseau italien est en cous d'étude".
Ces interconnections permettent un approvisionnement plus fiable et une meilleure gestion de la production.
En effet, lorsque la consommation interne est dans le seuil bas, l'excédent d'énergie est exporté vers les pays voisins ou dans le futur vers l'étranger "l'Europe".
Ces interconnections malgré leur intérêt économique, posent un ensemble de problèmes techniques parmi les quels les problèmes liés à la circulations d'informations sur le réseau suite à l'utilisation du système appelé : "Télécommande centralisée : TC".

Télécommande centralisée : TC

Les distributeurs d'énergie électrique disposent dans leur réseau d'un système permettant de commander à distance et avec un minimum de frais d'installation les différents types de consommateurs d'énergie.
Chaque distributeur établit un programme complet de télécommande, en fonction de sa courbe de charge.
Ce dispositif ou système porte le nom de "**TELECOMMANDE CENTRALISEE**" et il a pour but, par exemple, de :

commuter les compteurs d'énergie entre le tarif haut et le tarif bas.
bloquer les chauffages électriques dans les immeubles grand public.
bloquer les fours industriels.
enclencher l'éclairage public.
d'autres fonctions sont également possibles.

Les réseaux de distribution d'énergie électrique sont interconnectés. Ils disposent tous de télécommande centralisée.
Vu la multitude de réseaux, il est nécessaire de **BLOQUER** les différentes fréquences, en amont, de leur point d'injection. Si cette précaution n'est pas prise, il peut se produire des démarrages intempestifs de télécommande à un moment et à un endroit non désiré (sorte de pollution des réseaux pouvant provoquer des dégâts).
Pour empêcher les fréquences télécommandes de remonter dans l'interconnexion, on place au point de connexion un circuit électrique spécial, composé de resistor, bobine et capacité.
Les constituants de ce circuit sont disposés de façon à former un bouchon pour les fréquences indésirables.

Le schéma de principe est le suivant :

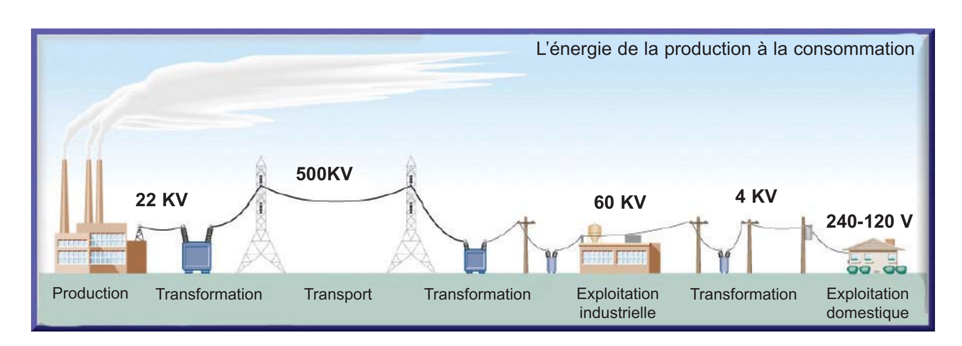
Ces différentes configurations seront traitées en détails plus bas dans ce cours.

B- RAPPEL

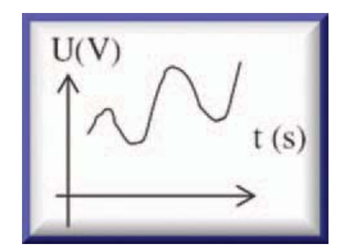
1- Quelques notions

1-1 L'énergie de la production à la consommation

Le principal fournisseur d'énergie sous forme de courant alternatif sinusoïdal en Tunisie est la S.T.E.G "société tunisienne d'électricité et de gaz ".

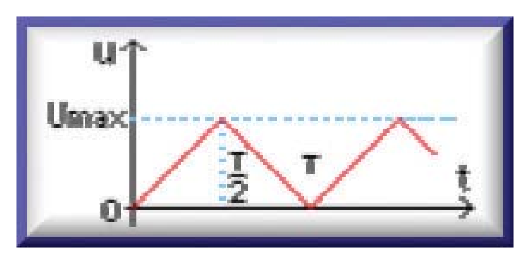


1-2 Courants variables :



En régime variable, les courants et les tensions sont des grandeurs variant avec le temps.

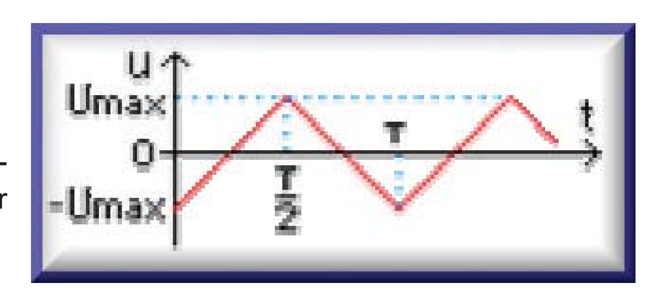
1-3 Courants périodiques :



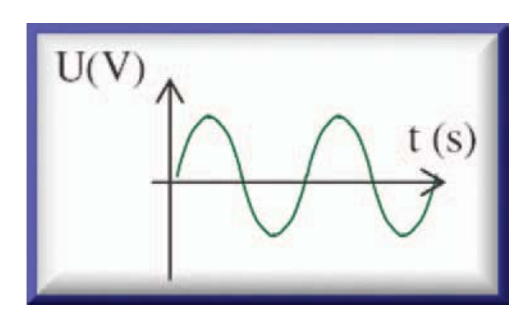
Un courant est périodique lorsqu'il se reproduit identiquement à lui-même pendant une même durée.

1-4 Courants alternatifs :

Un courant alternatif prend des valeurs tantôt positives, tantôt négatives. Sa valeur moyenne est nulle.



1-5 Courants sinusoïdaux :



Un courant sinusoïdal est un courant alternatif symétrique dont la représentation graphique est une sinusoïde.

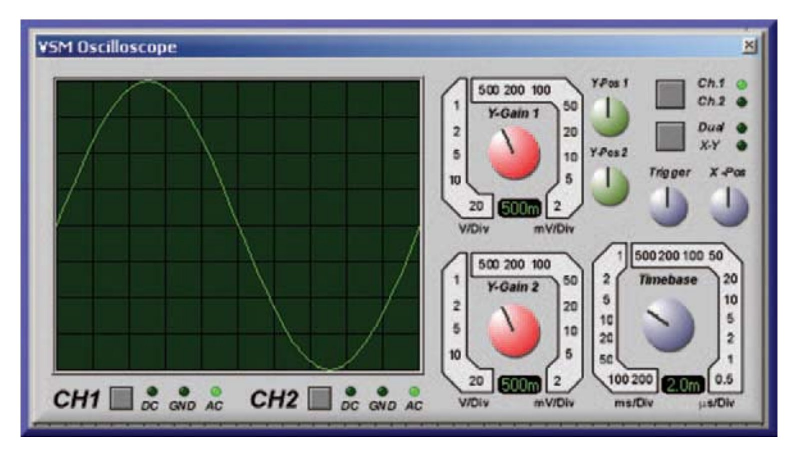
1-6 Dipôle :

On appelle dipôle, toute portion de circuit comprise entre 2 bornes (pôles). Un dipôle peut comporter n'importe quel composant ou appareil électrique: résistor, bobine, condensateur, moteur, générateur ...

2- Les caractéristiques du courant alternatif sinusoïdal :

2-1 Introduction :

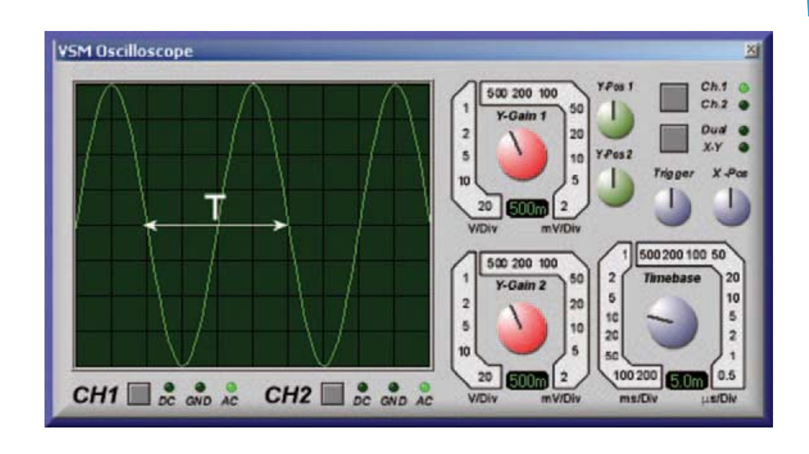
Un courant alternatif sinusoïdal est un courant variant périodiquement en s'inversant. Sa représentation graphique en fonction du temps est une sinusoïde.



2-2 Caractéristiques :

a/ Période :

Un phénomène est périodique s'il se reproduit identiquement pendant des intervalles de temps égaux appelés périodes.
La période est donc une durée.
Le symbole de la période est T.
Exemple : rotation de la Terre sur elle-même (T = 24 h ou 86 400 s).

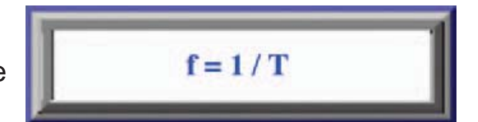


b/ Fréquence :

La fréquence d'un phénomène périodique est égale au nombre de périodes par seconde.
Le symbole de la fréquence est f. L'unité légale est le hertz dont le symbole est Hz.
Si un phénomène se reproduit identiquement toutes les 40 ms :

- ce phénomène est périodique ;
- la période est T = 40 ms = 0,04 s = 40.10^{-3} s
- la fréquence est f = 1 / T = 1 / 0,04 = 25 Hz

Avec "f" en Hz et "T" en secondes
Une fréquence de 1000 Hz signifie que le phénomène se reproduit identiquement 1000 fois en 1 s.
En Tunisie, la fréquence du courant électrique délivrée par la STEG est de 50 Hz.



c/ Valeur maximale et valeur efficace :

- Vecteur de Fresnel :

La sinusoïde représentant la tension alternative u ou l'image du courant i, est engendrée par le vecteur $\vec{U}_m$ / $\vec{I}_m$ appelé vecteur de Fresnel.

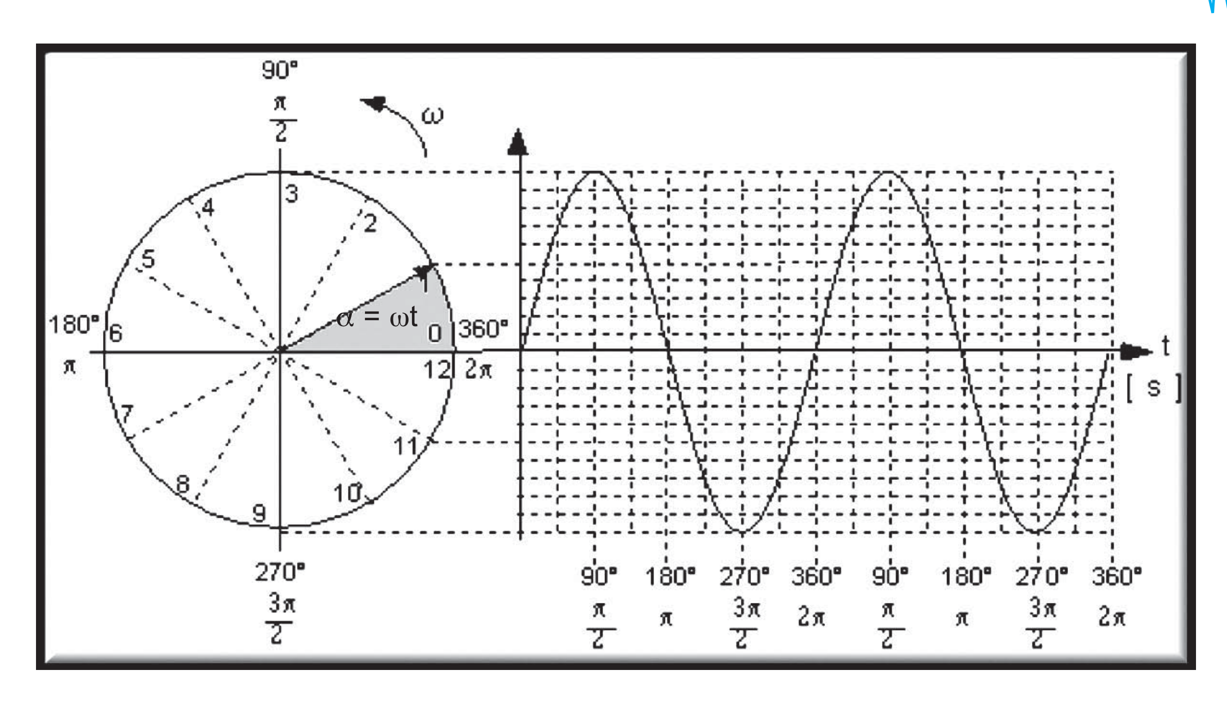
Dans le cas vu plus haut, à l'instant t = 0 , u = 0 / i = 0.
L'axe horizontal (Ox) est appelé origine des phases. T est la période de la tension ou du courant ; elle correspond au temps mis par le vecteur $\vec{U}_m$ ou $\vec{I}_m$ pour accomplir 1 tour.
Le module du vecteur de Fresnel représente la valeur maximale de la tension ou du courant.

Tension efficace :

$$U_{eff} = \frac{U_m}{\sqrt{2}}$$

Intensité efficace :

$$I_{eff} = \frac{I_m}{\sqrt{2}}$$



d/ La pulsation :

Le vecteur tournant $\vec{U}_m$ est en mouvement circulaire uniforme.
Soit α l'angle balayé pendant le temps t ; la vitesse angulaire ω du vecteur tournant est exprimée par la relation : $\omega = \alpha / t$.
Avec α en radians, t en secondes et ω en $rad.s^{-1}$
ω est appelée la pulsation angulaire.
2π est l'angle balayé pendant le temps d'une période T, donc : $\omega = 2\pi / T$
D'autre part, f = 1/T. La relation peut donc s'écrire :

$$\omega = 2\pi f$$

e/ Valeur instantanée :

La sinusoïde vue précédemment représente l'allure d'une tension alternative sinusoïdale ou l'image d'un courant alternatif sinusoïdal.
Cette représentation est une fonction sinusoïdale du temps :

t ==> u(t) et t ==> i(t)

u et i sont les valeurs instantanées de la tension et du courant.
Les expressions mathématiques "générales" de telles grandeurs sont les suivantes :

$$u(t) = U_m \sin(\alpha + \varphi') \qquad i(t) = I_m \sin(\alpha + \varphi)$$

Or $\alpha = \omega t$

d'où : $u(t) = U_m \sin(\omega t + \varphi') = U\sqrt{2}\sin(\omega t + \varphi')$ $\qquad i(t) = I_m \sin(\omega t + \varphi) = I\sqrt{2}\sin(\omega t + \varphi)$

Avec :
- Um ou Im : la valeur maximale.
- U ou I : la valeur efficace.

u	ω	t	ωt
en volts	en rad/s	en secondes	en radians

- Cas où u(t) = 0 ou i(t) = 0 pour " t = 0 "

$$u(t) = U_m \sin(\omega t + \varphi') = U\sqrt{2}\sin(\omega t + \varphi') = 0 \implies \sin\varphi' = 0 \implies \varphi' = 0$$

D'où l'expression de u : $u(t) = U_m \sin(\omega t) = U\sqrt{2}\sin(\omega t)$

NB : la même démarche s'applique pour **"i(t)"**.

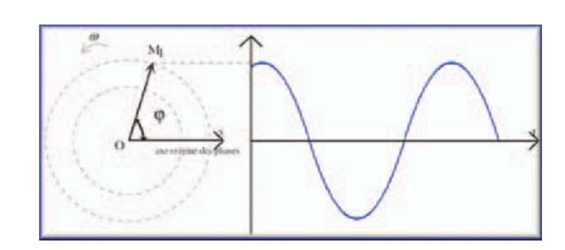
- Cas où u(t) ≠ 0 ou i(t) ≠ 0 pour " t = 0 "

Dans ce cas l'expression instantanée garde la forme suivante :

$$u(t) = U_m \sin(\omega t + \varphi')$$

$$i(t) = I_m \sin(\omega t + \varphi)$$

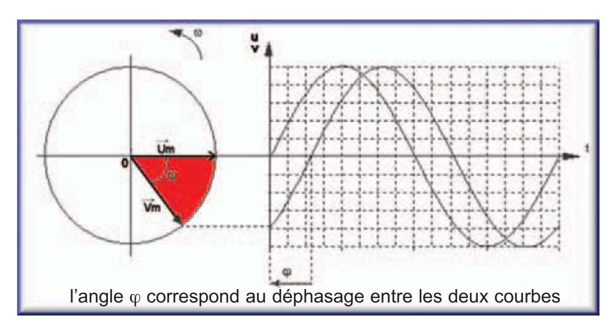
φ est appelé phase à l'origine



f/ Notion de déphasage :

Exemple : déphasage entre deux tensions.





l'angle φ correspond au déphasage entre les deux courbes

- Les vecteurs $\vec{U}_m$ et $\vec{V}_m$ représentent respectivement les tensions **"u"** et **"v"**.
- On appelle déphasage entre u et v l'angle : $\varphi = (\vec{U}_m, \vec{V}_m)$
- La tension **v** est en retard sur la tension **u**.

C- DIPOLES SOUS UNE TENSION ALTERNATIVE MONOPHASEE

1- comportement des dipôles élémentaires en courant continu :

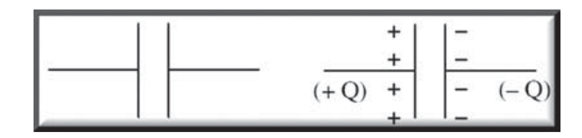
1-1 Loi d'ohm : Le comportement d'une portion de circuit ne contenant pas de composants actifs ne dépend que de sa résistance. Un tel circuit est dit purement résistif et l'on a la relation **U = R I**.

1-2 Rôle d'une bobine :

La grandeur caractéristique d'une bobine est l'inductance notée L. Elle s'exprime en henrys (H). En courant continu, la bobine joue le rôle d'un interrupteur fermé.
Une bobine ne manifeste ses effets qu'en régime variable (fermeture du circuit par exemple).



1-3 Rôle d'un condensateur : Un condensateur est un dipôle constitué de deux armatures conductrices séparées par un isolant (le diélectrique). Lorsqu'on applique une tension U aux bornes d'un condensateur, les armatures emmagasinent des charges électriques, en égale quantité Q et de signe opposé : on dit que le condensateur est chargé. La quantité d'électricité Q accumulée sur une armature est proportionnelle à la tension U appliquée. Le coefficient de proportionnalité C est appelé capacité du condensateur.
Q = C.U (Q : en coulombs, **U** : en volts, **C** en farads)
La capacité d'un condensateur dépend de sa géométrie et des caractéristiques électriques de son diélectrique.
Si un condensateur est dans un circuit, il est traversé par un courant pendant une durée assez brève, de sa charge ou de sa décharge, puis il se comporte comme une coupure dans le circuit et le courant dans celui-ci devient nul.



2- Comportement des dipôles élémentaires en courant alternatif :

2-1 Introduction : En alternatif, la loi d'ohm **u = Ri** n'est en général plus vérifiée car, outre les récepteurs purement résistifs, les récepteurs inductifs et capacitifs vont se manifester en permanence puisque la tension appliquée et le courant qui en résulte vont varier en permanence.

2-2 Rôle d'une inductance : elle se comporte comme un élément actif et contribue à fixer la valeur du courant (développement d'une f.é.m. auto induite). Une bobine n'a jamais une inductance pure, toute bobine possède une résistance.

2-3 Rôle d'un condensateur : les bornes d'un condensateur changent constamment de polarité, le condensateur est donc toujours, soit en cours de charge, soit en cours de décharge : il en résulte qu'une portion de circuit est traversée en permanence par un courant même si elle comporte un condensateur, ce dernier ne constitue pas une coupure.

NB : Ces différentes constatations nous amènent à évoquer la notion d'impédance.

3- Impédance des dipôles passifs :

En régime sinusoïdal l'impédance **Z**, d'un dipôle passif et linéaire, est égale au quotient de la valeur efficace de la tension appliquée par la valeur efficace du courant qui le traverse :

$Z = \frac{U}{I}$	**U** : tension efficace en volts **I** : intensité efficace en ampères **Z** : impédance en ohms

NB : L'impédance d'un dipôle passif est linéaire et ne dépend que de la fréquence de la tension appliquée à ses bornes.

3-1 Dipôles inductifs : formés principalement de bobines, dont les caractéristiques sont la résistance interne du bobinage "r" mesurée en Ω (Ohm) et l'inductance "L" mesurée en H (Henry).
En négligeant "r" le dipôle est dit parfait et l'impédance s'écrit alors : $X_L = L\,\omega = 2\pi\,L\,f$

Avec : X_L en ohms ; L en henrys ; ω en radians par seconde ; f en hertz

3-2 Dipôles capacitifs : formés principalement de condensateurs, dont la capacité dépend de ses caractéristiques géométriques et de la nature du diélectrique employé. L'impédance dans ce cas prend la forme suivante :

$$X_C = \frac{1}{C\omega}$$

Avec : **X_C ou Z en ohms, C en farads, ω en radians par seconde**

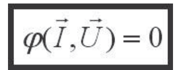
3-2 Dipôles resistifs : formés principalement d'éléments passifs "résistors".
L'impédance dans ce cas est égale à la résistance du résistor.

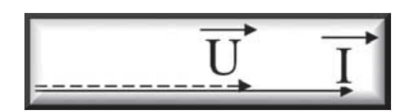
$$Z = R \longrightarrow U = RI = ZI$$

4- Déphasages dans les dipôles passifs :

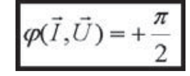
4-1 Introduction : dans tous dipôle soumis à une tension "u" et parcouru par un courant alternatif "i" sinusoïdal de pulsation ω, un déphasage φ prendra naissance entre l'intensité "i" et la tension "u".
Dans ce qui suit on prendra l'intensité comme origine des phases. Le sens positif des phases est le sens trigonométrique, l'angle de déphasage φ doit être exprimé en radians.

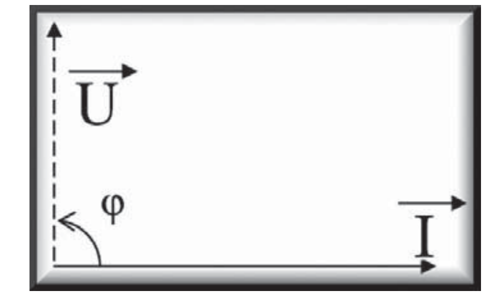
4-2 Cas d'un résistor pur : Dans ce cas "u" et "i" sont en phase,


$\varphi(\vec{I},\vec{U}) = 0$



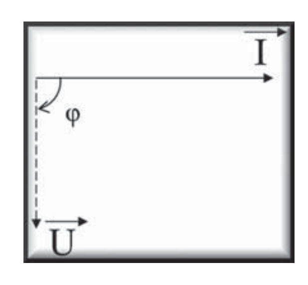
4-3 Cas d'une bobine pure : Dans ce cas "u" est en quadrature avance sur "i"quelles que soient :
- l'inductance L de la bobine ;
- la fréquence f du courant.


$\varphi(\vec{I},\vec{U}) = +\frac{\pi}{2}$



4-4 Cas d'une capacité pure : Dans ce cas "u" est en quadrature arrière sur "i" quelles que soient
- la capacité C du condensateur ;
- la fréquence f du courant.

$$\varphi(\vec{I},\vec{U}) = -\frac{\pi}{2}$$



D- ASSOCIATION DES DIPOLES EN REGIME SINUSOIDAL

1- Associations de dipôles en série :
en série, le courant est commun aux constituants du dipôle.
- sa valeur instantanée i est commune ;
- sa valeur efficace I est commune.

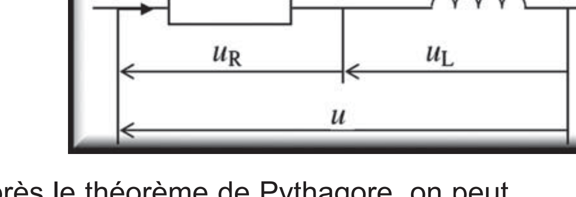
A chaque instant, la tension u entre les bornes de l'association est égale à la somme des tensions aux bornes de chacun des dipôles élémentaires.

Ceci permet d'écrire : $u = u_1 + u_2 + u_3 + ...$

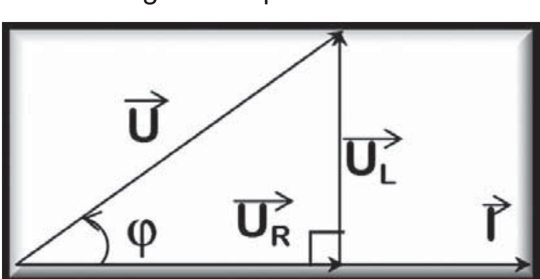
En régime sinusoïdal, on peut écrire la relation vectorielle suivante :

$$\vec{U} = \vec{U}_1 + \vec{U}_2 + \vec{U}_3 + ...$$

1-1 Circuit "R-L" série : Dans ce cas on peut écrire :
- $u = u_R + u_L$
- $\vec{U} = \vec{U}_R + \vec{U}_L$



D'où le triangle des tensions, en prenant $\vec{I}$ comme origine des phases.



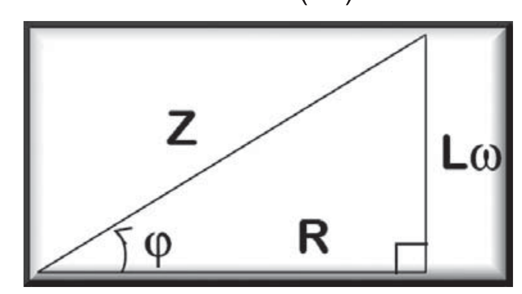
D'après le théorème de Pythagore, on peut écrire : $U^2 = U^2_R + U^2_L$

Or $U = ZI$; $U_R = RI$ et $U_L = L\omega I$

D'où $(ZI)^2 = (RI)^2 + (L\omega I)^2$

En divisant l'égalité précédente par la quantité I^2 on obtient : $Z^2 = R^2 + (L\omega)^2$

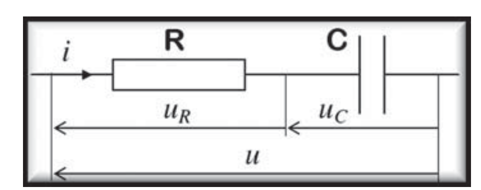
D'après cette relation, on peut tracer le triangle des impédances :



D'où :

Impédance : $Z = \sqrt{R^2 + (L\omega)^2}$ $\quad \cos\varphi = \frac{R}{Z}$

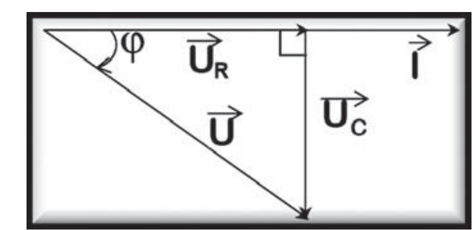
1-2 Circuit "R-C" série :



Comme précédemment, on peut écrire :

- $U = U_R + U_C$
- $\vec{U} = \vec{U}_R + \vec{U}_C$

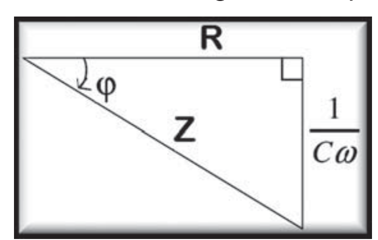
D'où le triangle des tensions, en prenant I comme origine des phases



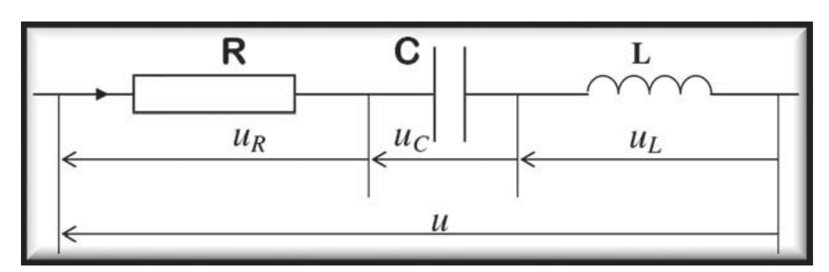
D'après le théorème de Pythagore,
on peut écrire : $U^2 = U^2_R + U^2_C$
Or $U = ZI$; $U_R = RI$ et $U_C = X_C I$
Avec $X_C = 1 / C\omega$ ==> $U_C = I / C\omega$
D'où $(ZI)^2 = (RI)^2 + (I / C\omega)^2$
divisons par la quantité I^2 on obtient :

$$Z^2 = R^2 + (1 / C\omega)^2 \quad ==> \quad Z = \sqrt{R^2 + \left(\frac{1}{C\omega}\right)^2}$$

D'après cette relation, on peut tracer le triangle des impédances



1-3 Circuit "R-L-C" série :

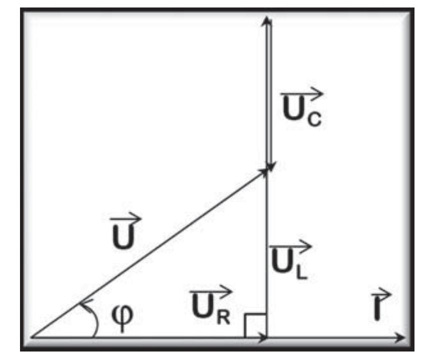


Dans ce cas on peut écrire :

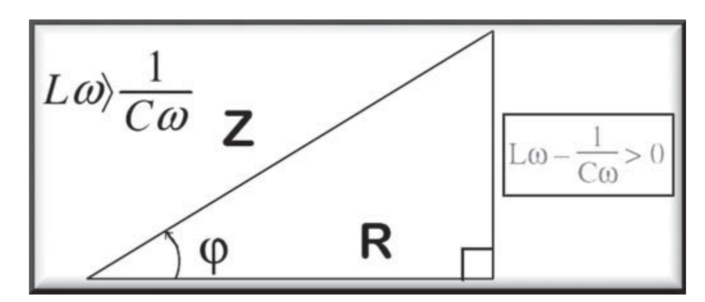
- $u = u_R + u_C + u_L$
- $\vec{U} = \vec{U}_R + \vec{U}_C + \vec{U}_L$

Pour tracer le triangle des tensions, trois cas sont à prévoir en fonction des valeurs de U_L et U_C.

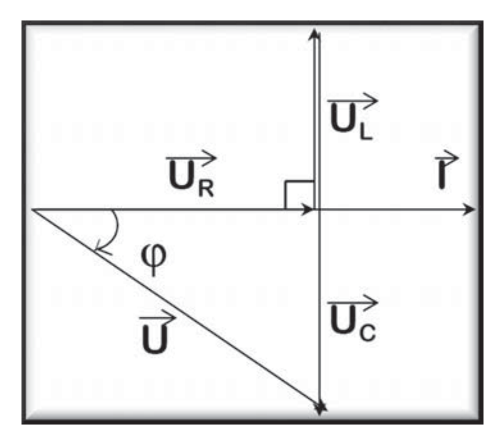
a- Cas de $U_L > U_C$



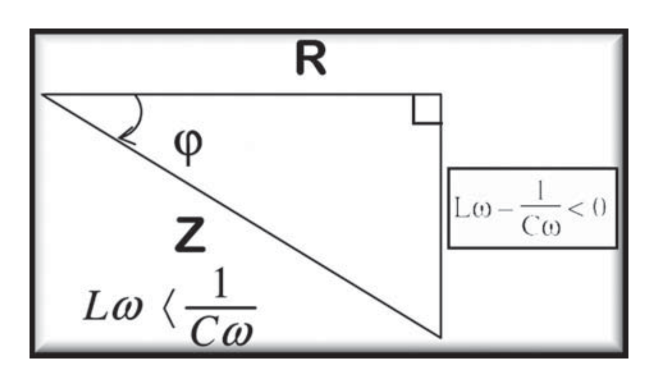
D'où le triangle des impédances



b- Cas de $U_L < U_C$



D'où le triangle des impédances



Sachant que :

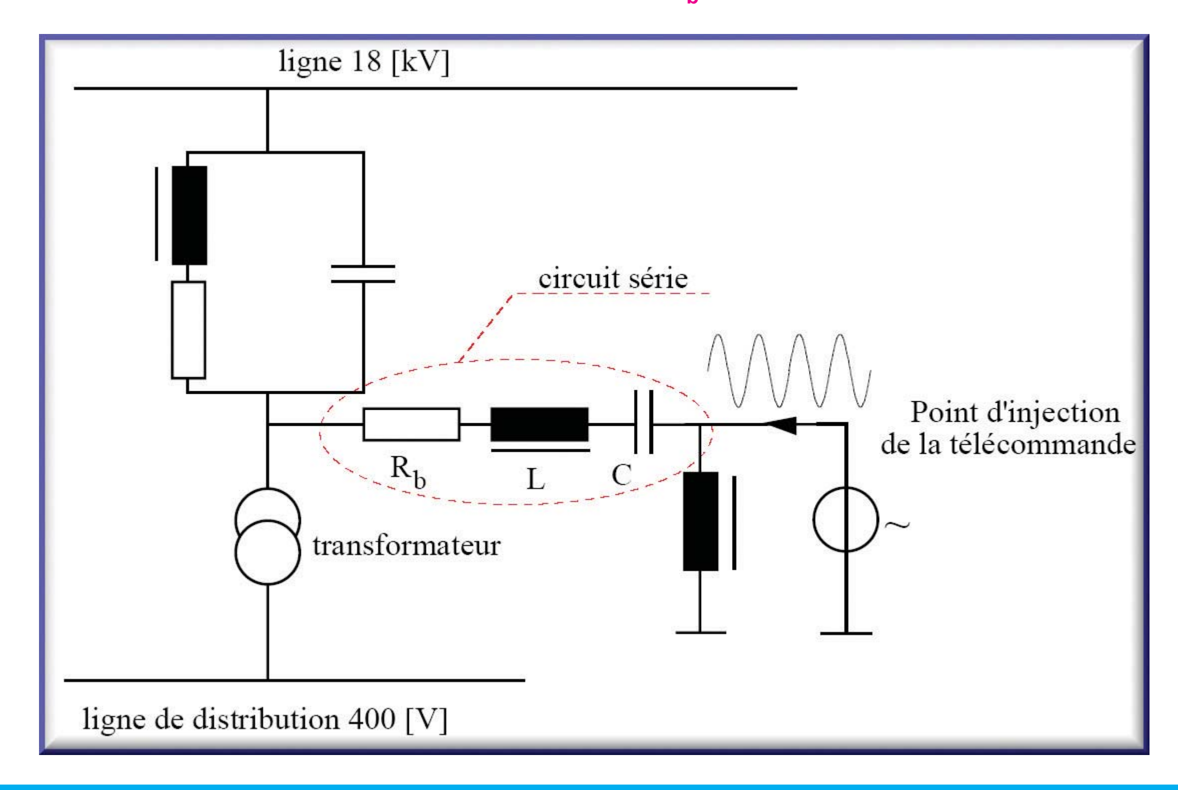
$$U_L = L\omega I \;;\; U_R = RI \;;\; U_C = I / C\omega$$

Si on applique le théorème de Pythagore dans le triangle des impédances, on obtient :

$$Z^2 = R^2 + (L\omega - 1 / C\omega)^2$$

D'où $Z = \sqrt{R^2 + (L\omega - \frac{1}{C\omega})^2}$ et $\cos\varphi = \frac{R}{Z}$

c- Cas de $U_L = U_C$: On revient au cas du circuit de télécommande centralisée et on s'intéresse à la configuration série des constituants "**R_b**, **L** et **C**".



Ce circuit est placé dans le but d'empêcher le passage de la fréquence réseau "**50Hz**" vers le générateur de fréquences de télécommande ; par contre les fréquences relatives aux ordres de télécommande doivent pouvoir transiter vers le transformateur et la ligne de distribution associée.

Problème technique : comment dimensionner les constituants de ce circuit pour : √ bloquer la fréquence réseau de **50Hz**, (cette fréquence est appelée fréquence de résonance elle est notée : "f_0") √ laisser passer la fréquence de télécommande ?

NB : Ces éléments doivent être dimensionnés pour supporter le courant nominal I_{nom} et le courant de court-circuit I_{cc} du réseau.

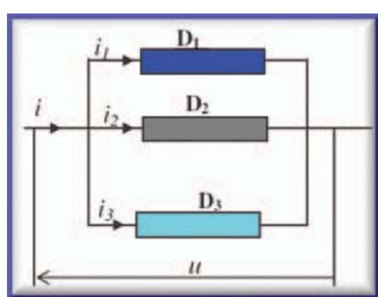
Théorie de la résonance du circuit série :
Une inductance **L** et un condensateur **C** sont montés en série. Ils sont soumis à la tension **U** avec une fréquence **f**. Selon leurs caractéristiques, ces éléments auront une certaine impédance totale **Z** à la fréquence du réseau **50 [Hz]**. Si la fréquence change, l'impédance totale **Z** se trouvera alors modifiée.

Activité
Réaliser l'activité pratique **N°1** du TP **B11-1** dans le manuel d'activités : cette activité consiste à déterminer les différentes caractéristiques d'un circuit résonnant série.

Conclusion :

Dans ce cas on dit qu'on est en **"résonance série"**

2- Associations de dipôles en parallèle ou en dérivation :



En dérivation, la tension est commune aux dipôles
- sa valeur instantanée u est commune ;
- sa valeur efficace U est commune.

A chaque instant, l'intensité i du courant dans le circuit principal est égale à la somme des intensités des courants dérivés.
Ceci permet d'écrire : $i = i_1 + i_2 + i_3 + ...$

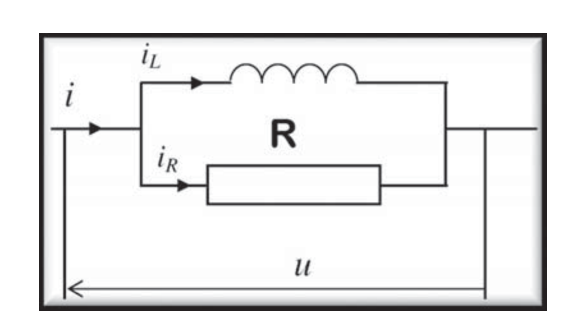
En régime sinusoïdal, la relation vectorielle est : $\vec{I} = \vec{I}_1 + \vec{I}_2 + \vec{I}_3 + ...$

2-1 Circuit "R-L" dérivation :

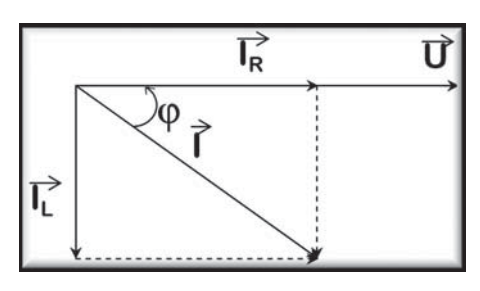
Dans ce cas on a :

- $I_R = \dfrac{U}{R}$ avec $\vec{I}_R$ en phase avec $\vec{U}$
- $I_L = \dfrac{U}{L\omega}$ avec $\vec{I}_L$ en quadrature

arrière sur $\vec{U}$.



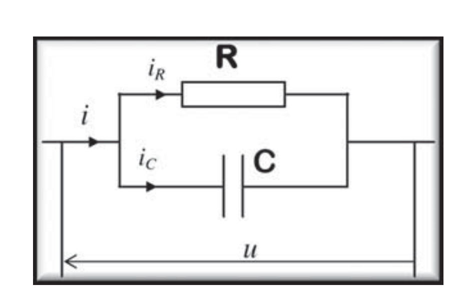
D'où le triangle des courants, en prenant $\vec{U}$ comme origine des phases



Le triangle formé par $(\vec{I};\ \vec{I}_R;\ \vec{I}_L)$ est un triangle rectangle donc :

- $I^2 = I_R^2 + I_L^2$ d'où $tg\varphi = \frac{I_L}{I_R} = (\frac{U}{L\omega})/(\frac{U}{R}) = \frac{U}{L\omega}$
- et **$\varphi > 0$.**

2-2 Circuit "R-C" dérivation :

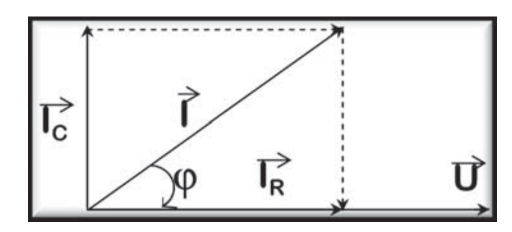


Dans ce cas on a :

- $I_R = \frac{U}{R}$ avec $\vec{I}_R$ en phase avec $\vec{U}$
- $I_C = UC\omega$ avec $\vec{I}_C$ en quadrature

avant sur $\vec{U}$

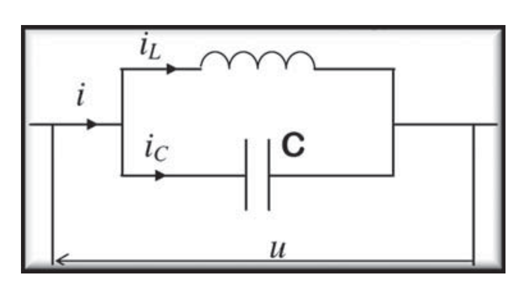
D'où le triangle des courants, en prenant $\vec{U}$ comme origine des phases.



Le triangle formé par $(\vec{I};\ \vec{I}_R;\ \vec{I}_C)$ est un triangle

- $I^2 = I_R^{\ 2} + I_C^{\ 2}$ d'où $tg\varphi = \frac{I_C}{I_R} = (UC\omega)/(\frac{U}{R}) = RC\omega$
- et **$\varphi < 0$**

2-3 Circuit "L-C" dérivation :



Dans ce cas on a :

- $I_C = UC\omega$ avec $\vec{I}_C$ en quadrature

avant sur $\vec{U}$.

- $I_L = \frac{U}{L\omega}$ avec $\vec{I}_L$ en quadrature

arrière sur $\vec{U}$.

- $\vec{I}_C$ et $\vec{I}_L$ sont en opposition de phase.

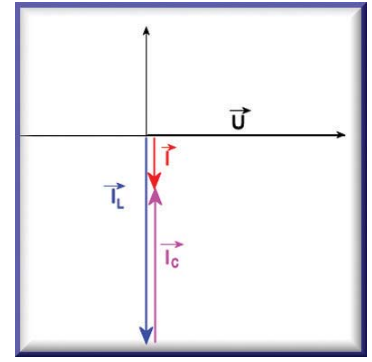
Cette configuration permet d'étudier 3 cas possibles :



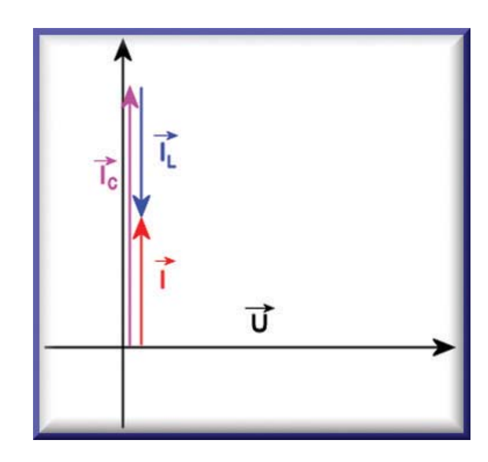
a/ Cas de $I_L > I_C$

- $\vec{I}$ a le même sens que $\vec{I}_L$, il est donc en quadrature

arrière sur $\vec{U}$.



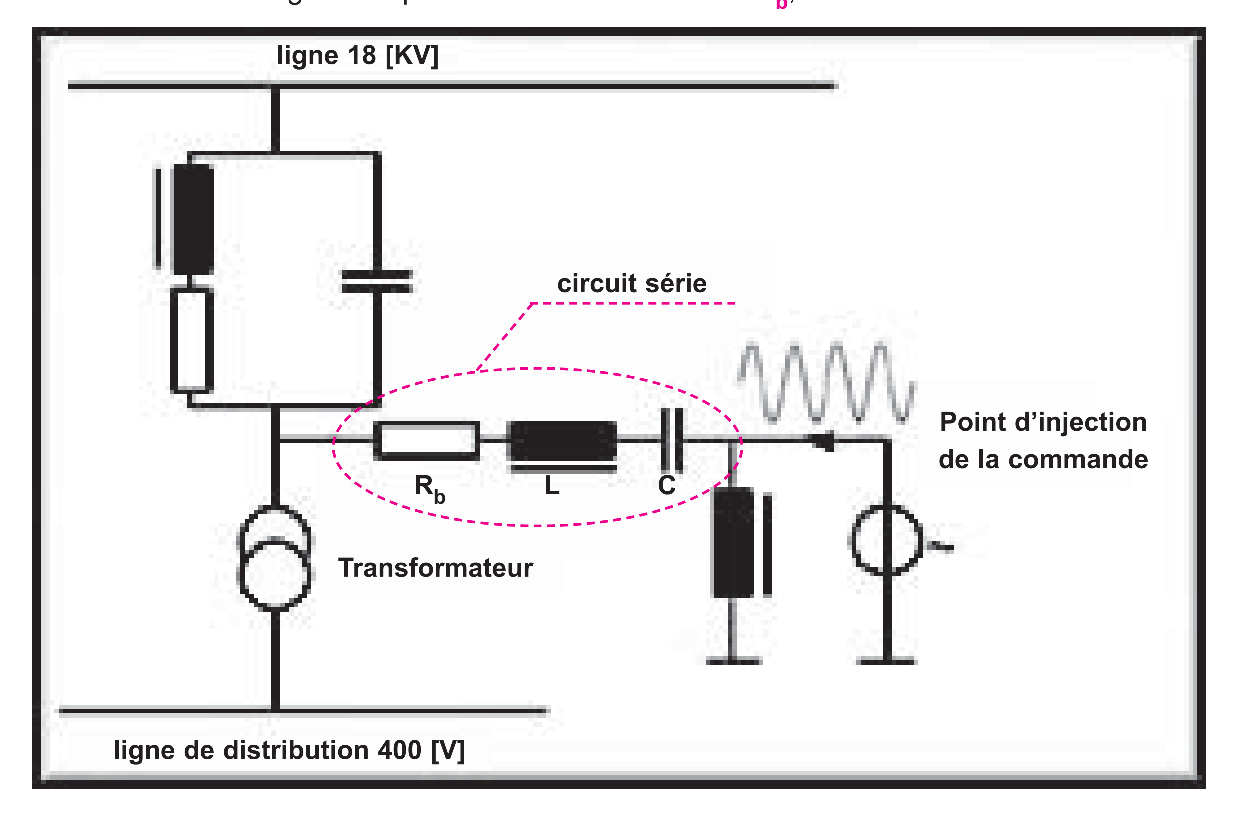
- $\varphi(\vec{I},\vec{U}) = +\frac{\pi}{2}$
- Le dipôle ainsi constitué est globalement inductif.
- $\vec{I}$ est en retard sur $\vec{U}$.



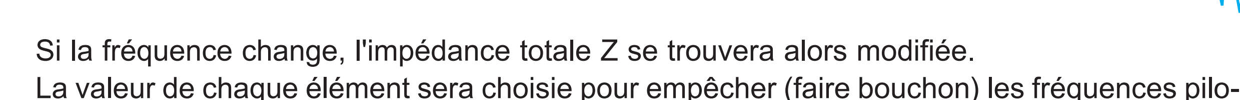
b/ Cas de $I_L < I_C$

- $\vec{I}$ a le même sens que $\vec{I}_C$, il est donc en quadrature avant sur $\vec{U}$.
- $\varphi(\vec{I},\vec{U}) = -\frac{\pi}{2}$
- Le dipôle ainsi constitué est globalement capacitif.
- $\vec{I}$ est en avance sur $\vec{U}$.

c/ Cas de $I_L = I_C$: On reprend le cas du circuit de télécommande centralisée et on s'intéresse à la configuration parallèle des constituants "**R_b**, **L** et **C**".



Ce circuit est placé dans le but d'empêcher le passage de la ou des fréquences de télécommande vers la zone d'interconnexion "18KV", l'inductance L et le condensateur C sont montés en parallèle. Ils sont soumis à la tension U avec une fréquence f. Selon leurs caractéristiques, ces éléments auront une certaine impédance totale Z à la fréquence du réseau 50 [Hz]. Cette impédance devra être faible pour ne pas limiter le courant vers le transformateur.



Si la fréquence change, l'impédance totale Z se trouvera alors modifiée.
La valeur de chaque élément sera choisie pour empêcher (faire bouchon) les fréquences pilotes (télécommande) d'arriver sur la ligne 18 kV.

Problème technique : comment dimensionner les constituants de ce circuit pour
- bloquer la fréquence de télécommande
- laisser passer la fréquence de tréseau de 50Hz ?

NB : Ces éléments doivent être dimensionnés pour supporter le courant nominal I_{nom} et le courant de court-circuit I_{cc} du réseau.

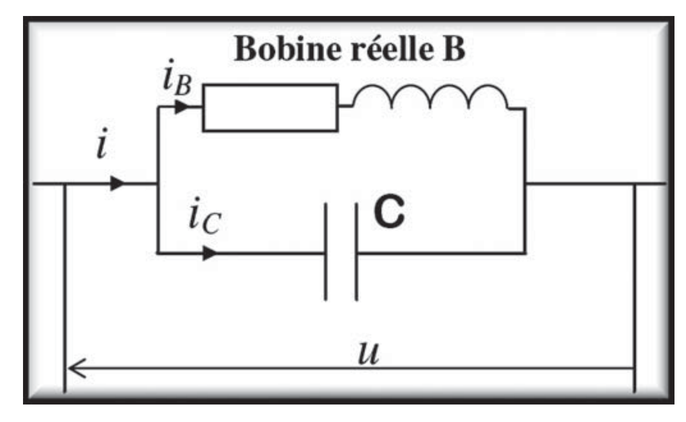
Activité
Réaliser l'activité pratique N°2 du TP B11-1 dans le manuel d'activités : cette activité consiste à déterminer les différentes caractéristiques d'un circuit bouchon.

- Lorsque $I_C = I_L$ on dit qu'on est en circuit bouchon idéal.

$$I_C = I_L \Rightarrow \frac{1}{L\omega} = C\omega \Rightarrow LC\omega_0^2 = 1 \Rightarrow \omega_0 = \sqrt{1/LC}$$

$\vec{I}_C$ et $\vec{I}_L$ sont en opposition de phase et malgré la présence d'une tension d'alimentation non nulle, le courant "I" est nul ; on dit que le circuit est bouchon.

- Circuit bouchon réel : le circuit idéal est purement théorique, dans la réalité les bobines ne sont pas dépourvues de résistances.



Quand la relation $LC\omega^2$ est satisfaite, les deux courants i_B et i_C ont pratiquement la même valeur efficace mais ne sont pas tout à fait en opposition de phase ($\vec{I}_C$ est en quadrature avant sur $\vec{U}$ mais $\vec{I}_L$ n'est pas tout à fait en quadrature arrière sur $\vec{U}$). Il s'en suit que le courant résultant n'est pas strictement nul, néanmoins sa valeur reste très faible; de plus il est pratiquement en phase avec $\vec{U}$.

E- RESOLUTION PAR LES NOMBRES COMPLEXES

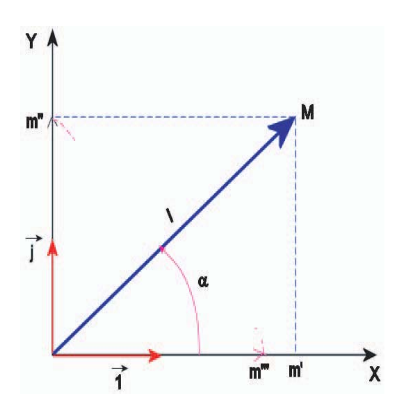
La méthode vectorielle évoquée précédemment permet de résoudre facilement les circuits dont la configuration est simple tel que le RLC série. Toutefois elle devient difficile à mettre en œuvre dès que la configuration du circuit commence à se compliquer. Dans ce cas on préfère tirer profit des possibilités offertes par l'emploi des nombres complexes pour résoudre ce type de problèmes.

Ecriture d'une grandeur sinusoïdale sous forme complexe

1- Introduction aux nombres complexes :

Considérons la fonction sinusoïdale suivante :

$$i(t) = I_m \sin(\omega t + \alpha) = I\sqrt{2}\sin(\omega t + \alpha)$$



de valeur efficace (**I**) et de pulsation ω . Sa représentation de Fresnel est un vecteur $\overrightarrow{OM}$ de module **I** déphasé de α par rapport à un axe **OX** pris comme référence et comme support pour toutes les fonctions en **sin ωt**, on aurait pu le prendre également en **cos ωt**.

✔ Première notation complexe : Un vecteur $\overrightarrow{OM}$ est parfaitement défini quand on a choisi deux axes rectangulaires de référence et quand on se donne soit, ses projections (**i'** et **i"**) sur les deux axes, soit son module **I** et son déphasage α par rapport à l'un des deux axes. Grâce aux notations imaginaires, on peut exprimer ce vecteur sous une forme algébrique. Pour cela on choisis un vecteur unité et on lui donne pour support l'axe **OX**. Par convention, on traduit une rotation de 90° du vecteur unité en multipliant ce vecteur par un nombre imaginaire **(j)**; un nombre capable de faire effectuer à un vecteur une rotation est encore appelé un opérateur. Ainsi l'axe **OY** est-il le support du vecteur $j \times \vec{1} = \vec{j}$ Après une nouvelle rotation de 90°, on obtient le vecteur $j^2 \times \vec{1}$ qui est en opposition avec le vecteur $\vec{1}$.
En définitive, on obtient la relation $\mathbf{j^2 = -1}$
Ainsi l'opérateur **(j)** qu'on a défini arbitrairement possède-t-il toutes les propriétés du nombre imaginaire $j = \sqrt{-1}$
Les axes **OX** et **OY** qui portent respectivement les vecteurs $\vec{i}$ et $\vec{j}$ sont appelés axe *réel* et axe *imaginaire*.
Soit **m'** et **m"** les projections de l'extrémité **M** sur les deux axes, on a : $\overrightarrow{OM} = \overrightarrow{Om'} + \overrightarrow{Om''}$
Le vecteur $\overrightarrow{Om'}$ qui a même support que le vecteur unité s'exprime alors par le nombre algébrique (**a'**), ce nombre étant positif si les deux vecteurs ont même sens et négatif s'ils sont de sens contraires.
Le vecteur $\overrightarrow{Om''}$ s'exprimerait, lui aussi, par le nombre algébrique (**a"**) si ce vecteur était porté par l'axe **OX** suivant $\overrightarrow{Om'''}$, mais comme ce vecteur $\overrightarrow{Om'''}$ doit tourner de **90°** pour donner le vecteur $\overrightarrow{Om''}$, l'expression de $\overrightarrow{Om''}$ sera **ji"**.

d'où la première notation complexe :

$$\underline{I} = i' + ji''$$

✔ Deuxième notation complexe : Calculons **i'** et **i"** en fonction du module **I** et du déphasage α , tous calcul fait, on obtient :

i' = I cos α , i" = I sin α d'où

$$\underline{I} = I(\cos\alpha + j\sin\alpha)$$

Cette même expression peut se mettre sous la forme suivante : $\mathbf{Ie^{j\alpha}}$
ou encore $\mathbf{I}\,\underline{/\alpha}$ mais cette fois avec α en degrés.

2- Application aux cas des dipôles élémentaires :

Considérons un courant sinusoïdal dont l'expression instantanée est :

$$i(t) = I_M \cos(\omega t + \varphi)$$

Nous remarquons qu'il est possible d'associer à ce courant réel un courant fictif complexe, noté $\underline{I}$ et qui peut se mettre sou la forme : $\underline{I} = I_M(\cos(\omega t + \varphi) + j\sin(\omega t + \varphi))$ ou

$\underline{I} = I_M e^{j(\omega t + \varphi)}$. Il est très important de remarquer que cette transformation est simplement un outil mathématique qui facilitera les calculs par la suite.
$\underline{I}$ n'est en aucun cas le courant réel qui traverse le circuit. Pour trouver ce courant réel, il faut prendre la partie réelle de $\underline{I}$ si nous avons posé

$i(t) = I_M \cos(\omega t + \varphi)$

ou bien la partie imaginaire de $\underline{I}$ débarrassé de "j",

si nous avions posé $i(t) = I_M \sin(\omega t + \varphi)$
Nous pouvons opérer la même transformation pour la tension, en posant :

$$v(t) = VM\cos(\omega t + \varphi)$$

On obtient :

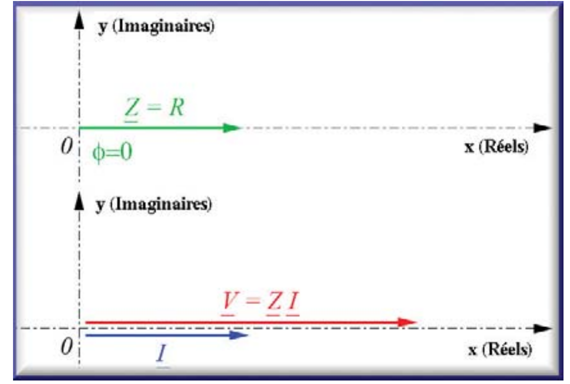
$$\underline{V} = V_M(\cos(\omega t + \varphi) + j\sin(\omega t + \varphi))$$

On en déduit que le module du nombre complexe $\underline{I}$ et $\underline{V}$ est l'intensité ou la tension maximale du courant ou de la tension réels et que l'argument de $\underline{I}$ et $\underline{V}$ est la phase du courant ou de la tension réels.
NB : Soit un nombre complexe **y = a + jb** ; le module de y vaut $\sqrt{a^2 + b^2}$

et l'argument de y vaut $\arctan g(\frac{b}{a})$

2-1 Résistor pur : nous avons vu que, dans ce cas, le courant qui traverse le résistor est en phase avec la tension
Si $i(t) = I_M \cos(\omega t)$; on a alors $v(t) = V_M \cos(\omega t)$

or $V_M = RI_M$ on peut donc écrire
sous la forme complexe : $\underline{V} = R\underline{I} = \underline{Z}\,\underline{I}$



d'où pour un résistor pur : $\underline{Z} = R$
"Z" s'appelle l'impédance du circuit. $\underline{Z}$ est l'impédance complexe du circuit.
Dans le cas d'un résistor $\underline{Z}$ est réel et vaut Z.

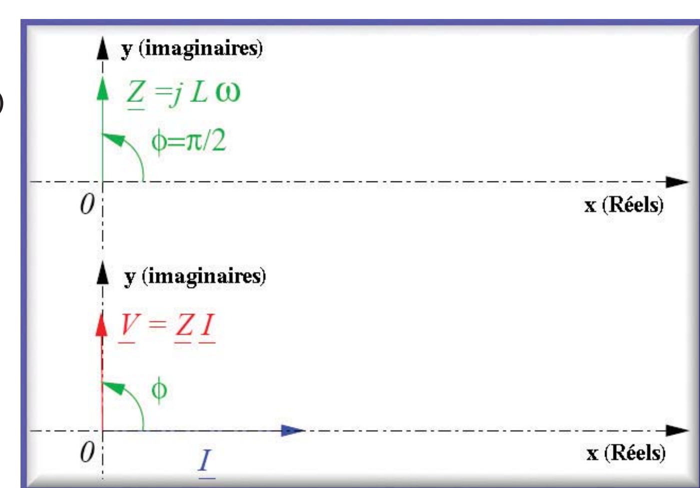
2-2 Inductance pure : Nous avons vu que la tension était en avance de $\frac{\pi}{2}$ sur le courant, c'est-à-dire que si l'on a : $i(t) = I_M \cos(\omega t)$

alors $v(t) = V_M \cos(\omega t + \frac{\pi}{2})$ or $V_M = L\omega I_M$ on peut donc écrire sous la forme

complexe : $\underline{I} = I_M(\cos\omega t + j\sin\omega t)$

donc :

$$\underline{V} = L\omega I_M(\cos(\omega t + \frac{\pi}{2}) + j\sin(\omega t + \frac{\pi}{2}))$$



sachant que

$$\cos(\omega t + \frac{\pi}{2}) = -\sin(\omega t)$$

et que

$$\sin(\omega t + \frac{\pi}{2}) = \cos(\omega t)$$

on obtient alors :

$$\underline{V} = L\omega I_M(-\sin\omega t + j\cos\omega t) = jL\omega I_M(\cos\omega t + j\sin\omega t)$$

sachant que $j^2 = -1$

donc V peut se mettre sous la forme suivante : $\underline{V} = jL\omega \underline{I} = \underline{Z}\,\underline{I}$

l'impédance complexe d'une inductance pure est donc : $\underline{Z} = jL\omega$

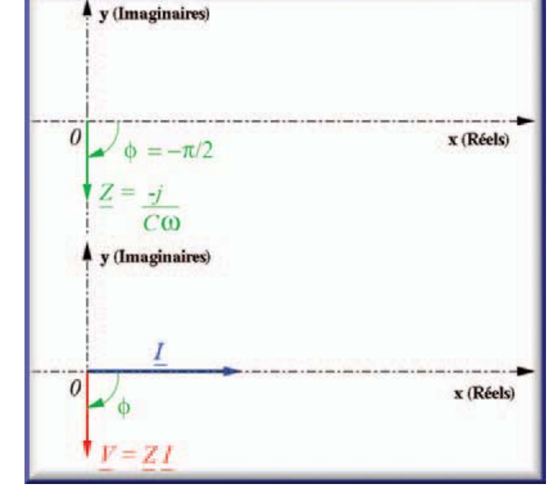
2-3 Capacité pure : Nous avons que la tension était en retard de $\frac{\pi}{2}$ sur le courant, c'est-à-dire si l'on a : $i(t) = I_M \cos(\omega t)$

alors $v(t) = V_M \cos(\omega t - \frac{\pi}{2})$ or $V_M = \frac{I_M}{C\omega}$ on peut donc écrire sous la forme

complexe : $\underline{I} = I_M(\cos\omega t + j\sin\omega t)$

donc :

$$\underline{V} = \frac{I_M}{C\omega}(\cos(\omega t - \frac{\pi}{2}) + j\sin(\omega t - \frac{\pi}{2}))$$



sachant que

$$\cos(\omega t - \frac{\pi}{2}) = \sin(\omega t)$$

et que

$$\sin(\omega t - \frac{\pi}{2}) = -\cos(\omega t)$$

on obtient alors :

$$\underline{V} = \frac{I_M}{C\omega}(\sin\omega t - j\cos\omega t) = -j\frac{I_M}{C\omega}(\cos\omega t + j\sin\omega t) = \underline{Z}\,\underline{I}$$

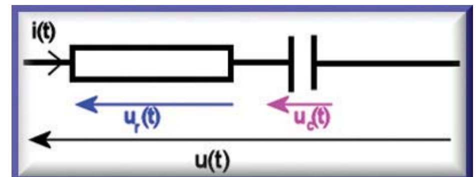
l'impédance d'une capacité pure est donc : $\underline{Z} = \frac{1}{jC\omega} = \frac{-j}{C\omega}$

2-4 Conclusion : Lorsqu'on passe en complexe dans un circuit alternatif, la différence de potentiel aux bornes d'un composant satisfait la loi d'Ohm complexe et l'association des impédances respecte les lois d'associations des résistors en continu sans perdre de vue que l'impédance est un nombre complexe.

- $\underline{V}$ différence de potentiel aux bornes de l'élément.
- $\underline{I}$ courant traversant cet élément.
- $\underline{Z} = R$ pour un résistor.
- $\underline{Z} = jL\omega$ pour une bobine.
- $\underline{Z} = \dfrac{1}{jC\omega} = \dfrac{-j}{C\omega}$ pour un condensateur.
- Pour des impédances en série : $\underline{Z}_{équivalante} = \underline{Z}_1 + \underline{Z}_2 + ...\underline{Z}_n$
- Pour des impédances en parallèle : $\dfrac{1}{\underline{Z}_{équivalente}} = \dfrac{1}{\underline{Z}_1} + \dfrac{1}{\underline{Z}_2} + ...\dfrac{1}{\underline{Z}_n}$

3- Applications :

3-1 Circuit "R-C" série : En appliquant la loi d'Ohm avec les grandeurs complexes associées, on a :



$\underline{U} = \underline{Z}\underline{I}$ l'impédance $\underline{Z}$ du circuit est la somme des impédances en série, soit : $\underline{Z} = R - \dfrac{j}{C\omega}$ d'où $\underline{U} = (R - \dfrac{j}{C\omega})\underline{I}$

le module de $\underline{U}$ est le produit des modules de $\underline{Z}$ et $\underline{I}$

==> $U = I\sqrt{R^2 + \dfrac{1}{C^2\omega^2}}$ alors

que le déphasage entre $\underline{U}$ et $\underline{I}$ est donné par : $tg\varphi = -\dfrac{1}{RC\omega}$



3-1 Circuit "R-L" parallèle : Appliquons la loi d'Ohm avec les grandeurs complexes associées.

$\underline{U} = \underline{Z}\underline{I}$ l'impédance $\underline{Z}$ du circuit parallèle s'obtient comme suit :

$\dfrac{1}{\underline{Z}} = \dfrac{1}{R} + \dfrac{1}{jL\omega}$ d'où $\underline{Z} = \dfrac{jRL\omega}{R + jL\omega}$ pour faciliter les calculs on doit mettre $\underline{Z}$ sous la forme



a + jb pour cela on doit multiplier le numérateur et le dénominateur par la quantité conjuguée du dénominateur ; d'où la nouvelle expression de $\underline{Z}$:

$$\underline{Z} = \frac{jRL\omega(R - jL\omega)}{(R + jL\omega)(R - jL\omega)} = \frac{RL^2\omega^2 + jR^2L\omega}{R^2 + L^2\omega^2}$$

Connaissant $\underline{Z}$ et $\underline{I}$ on peut déterminer l'expression de $\underline{U}$: $\underline{U} = \dfrac{RL^2\omega^2 + jR^2L\omega}{R^2 + L^2\omega^2}\underline{I}$

pour obtenir le module de $\underline{U}$, il suffit de multiplier le module de $\underline{I}$ par le module de $\underline{Z}$; le déphasage s'obtient en cherchant l'angle φ dont la tangente φ est obtenue.

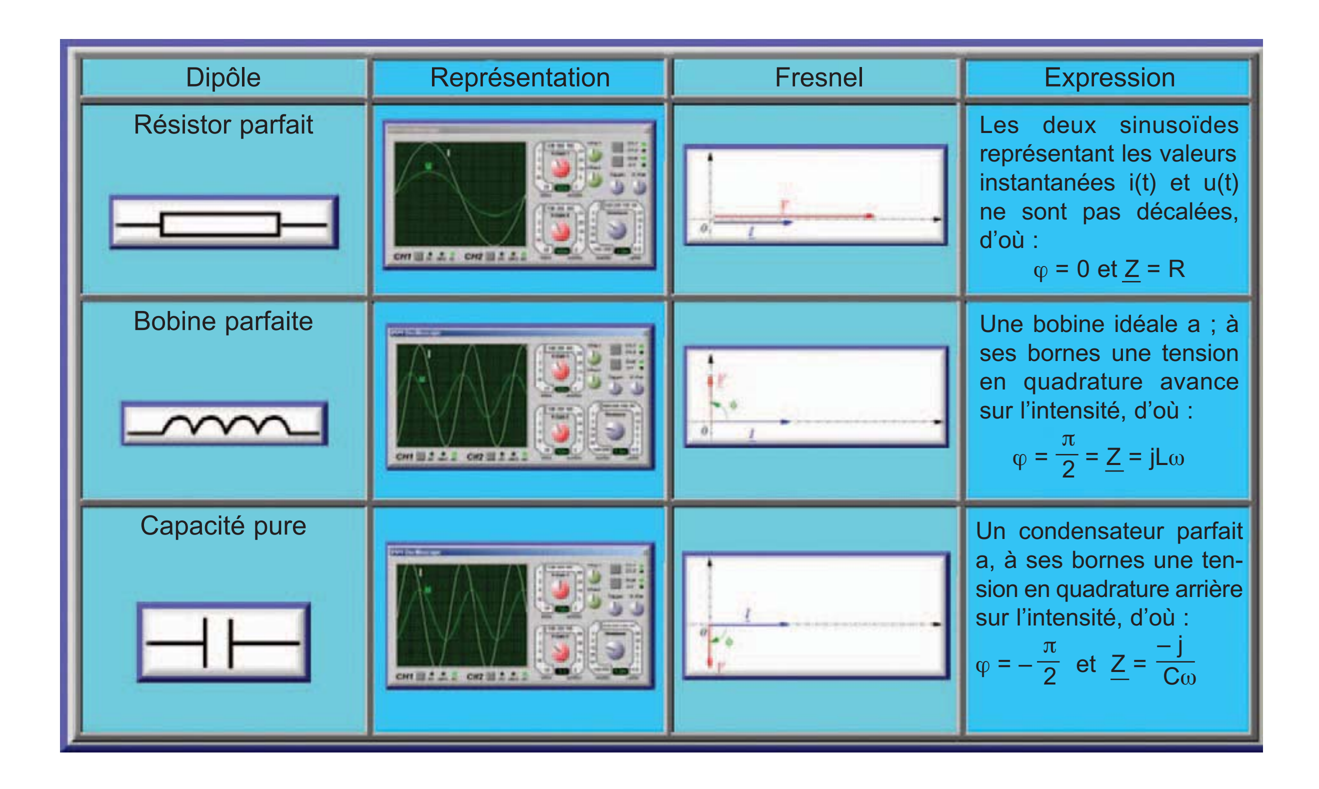
en divisant la partie réelle par la partie imaginaire de l'expression de $\underline{U}$; tous calcul fait on obtient Module de $\underline{U}$:

$$U = I\sqrt{\left(\frac{RL^2\omega^2}{R^2 + L^2\omega^2}\right)^2 + \left(\frac{R^2L\omega}{R^2 + L^2\omega^2}\right)^2}$$

Le déphasage entre $\underline{U}$ et $\underline{I}$ est donné par :

$$tg\varphi = \frac{R}{L\omega}$$

F- RESUMÉ



Dipôle	Représentation	Fresnel	Expression
Résistor parfait			Les deux sinusoïdes représentant les valeurs instantanées i(t) et u(t) ne sont pas décalées, d'où : $\varphi = 0$ et $\underline{Z} = R$
Bobine parfaite			Une bobine idéale a ; à ses bornes une tension en quadrature avance sur l'intensité, d'où : $\varphi = \frac{\pi}{2} = \underline{Z} = jL\omega$
Capacité pure			Un condensateur parfait a, à ses bornes une tension en quadrature arrière sur l'intensité, d'où : $\varphi = -\frac{\pi}{2}$ et $\underline{Z} = \frac{-j}{C\omega}$

G- EXERCICES

1- Exercice résolu :

Calculer l'impédance d'un dipôle constitué d'un résistor (R = 80ohms) et d'une Bobine parfaite (L = 1,2 H). Calculer la tension (f = 100 Hz) à leurs bornes (UR et UL) si l'intensité du courant dans le circuit est de 3A.

Correction :

$$Z = \sqrt{R^2 + (L\omega)^2} \quad ; \quad \omega = 2\pi f = 2\pi 100 = 628 rd^{-1}$$

$$L\omega = 1{,}2 \times 628 = 753{,}6\Omega$$

$$Z = \sqrt{(80)^2 + (753{,}6)^2} = 757{,}83\Omega$$

$$U_R = RI \Rightarrow U_R = 80 \times 3 = 240V$$

$$U_L = L\omega I \Rightarrow U_L = 753{,}6 \times 3 = 2260{,}8V \qquad U = ZI \Rightarrow U = 757{,}83 \times 3 = 2273{,}50V$$

D'où
$$U = \sqrt{U_R^2 + U_L^2} = \sqrt{(240)^2 + (2260{,}8)^2} = 2273{,}50V$$

2- Exercices à résoudre :

Exercice 01 :

Calculer l'impédance d'une bobine de résistance r = 12Ω et d'inductance L = 150 mH soumise à une tension U (220 V ; 125 Hz). En déduire la mesure de l'intensité efficace du courant traversant la bobine.

Exercice 02 :

On applique une tension de 220 V, 50 Hz entre les bornes d'un dipôle comportant un résistor R = 30Ω en série avec une bobine d'inductance L = 0,16 H.

- a/ Calculer l'impédance du dipôle ainsi constitué.
- b/ Calculer le courant traversant ce circuit.
- c/ Calculer le facteur de puissance.
- d/ Calculer la puissance absorbée.

Exercice 03 :

Une tension de 220 V, 50 Hz est appliquée à un dipôle comportant en série un condensateur de capacité C variable, une bobine d'inductance L = 0,7 H et un résistor R = 50Ω.

- a/ Calculer l'impédance du dipôle et l'intensité du courant pour les valeurs suivantes de C : 2 μF, 12μF et 20μF.
- b/ Calculer la valeur de C qui donne la résonance et la valeur que prend alors l'intensité du courant.

Exercice 04 :

Une tension sinusoïdale de valeur efficace U = 220 V, de fréquence 50 Hz, est appliquée à une bobine présentant une résistance.
R = 15Ω et une inductance L = 0,08 H.

a/ Calculer la réactance **XL** et l'impédance Z de la bobine.
b/ Calculer la valeur efficace I du courant la traversant.
c/ Calculer le déphasage j entre l'intensité du courant et la tension.
d/ Que deviennent ces résultats si la tension U devient U' = 110 V ?

Exercice 05 :

Un circuit électrique comporte en série un résistor de résistance R = 180Ω et une bobine (de résistance négligeable) d'inductance L = 400mH, est parcourue par un courant I = 0,5 A sous une fréquence f = 50 Hz.

a/ Calculer l'impédance Z du circuit ainsi constitué.
b/ Construire le diagramme de Fresnel relatif aux tensions.
c/ Déterminer le facteur de puissance cos φ de la portion de circuit ainsi constituée.

Exercice 06 :

Deux bobines B1 et B2 sont montées en dérivation. Entre leurs bornes communes on applique une tension de 120 V, 50 Hz. Sachant que R1= 50Ω L1 = 0,2 H, R2 = 40 ; L2 = 0,4 H.

a/ Calculer l'impédance de chacune des bobines.
b/ Calculer le courant dans chacune d'elles et les déphasages correspondants.
c/ Déterminer le courant total.
d/ Calculer l'impédance du circuit.

Exercice 07 :

Un condensateur C est monté en dérivation (C = 16µF) avec un résistor R = 200Ω.

a/ A quelle fréquence faut-il les alimenter pour que les courants $\mathbf{I_C}$ et $\mathbf{I_R}$ aient la même valeur efficace ?
b/ On applique à l'ensemble (C ; R) une tension U = 100 V de fréquence f ; calculer l'intensité du courant total, le déphasage de ce courant sur la tension et la valeur de l'impédance équivalente.

Exercice 08 :

Un bobine de résistance R = 20Ω et d'inductance L = 50 mH et un condensateur de capacité C = 60µF sont associés en dérivation sous une tension 220 V, 50 Hz.

a/ Déterminer les impédances Z_B et $\mathbf{Z_C}$ de la bobine et du condensateur.
b/ Déterminer la valeur efficace I_B et le déphasage B du courant dans la bobine.
c/ Déterminer la valeur efficace I_C et le déphasage C du courant dans le condensateur.
d/ Construire le diagramme de Fresnel : (pris comme origine des phases).
e/ Déterminer la valeur efficace **I** et le déphasage du courant principal.
f/ Déduire l'impédance Z du groupement.

Exercice 09 :

Une bobine (R = 10Ω , L = 36 mH) est soumise à une tension de 220 V, 50 Hz. Calculer

a/ Son impédance.

b/ L'intensité **I** du courant qui traverse la bobine

c/ Le déphasage entre le courant et la tension.

d/ On branche un condensateur de capacité 88μF en dérivation avec la bobine. Calculer :

* l'intensité I_c du courant dans le condensateur
* déterminer graphiquement la nouvelle intensité en ligne **I'**, ainsi que le nouveau facteur de puissance du groupement.

Exercice 10 :

On dispose d'un condensateur de 20μF, d'une bobine de résistance R=10Ω et de coefficient d'inductance propre L=0,3H, d'une tension sinusoïdale de valeur efficace 100V et de fréquence f=50hz.

Calculer l'intensité du courant et son déphasage par rapport à la tension quand on applique la tension successivement :

a/ Aux bornes du condensateur.

b/ Aux bornes de la bobine.

c/ A l'ensemble condensateur et bobine en série.

d/ A l'ensemble condensateur et bobine en parallèle.

Exercice 11 :

Une bobine B est alimentée en courant continu sous une tension U1 de 6V; elle est traversée par un courant I1 de 2A.

Lorsqu'on alimente cette même bobine en courant sinusoïdal de fréquence f=50Hz sous une tension efficace U de 110V, l'intensité I du courant est de 1A.

Calculer la résistance r et l'inductance L de cette bobine.

Exercice 12 :

Une bobine B (L=0,35H, r=3Ω) est placée en série avec une résistance non inductive R=52Ω et un condensateur C=10μF. On soumet l'ensemble à une tension sinusoïdale de valeur efficace U=110V dont on fait varier la fréquence f.

1- Calculer l'impédance Z de l'ensemble : bobine, résistance, condensateur pour f=100Hz ; quelle est alors l'intensité efficace I du courant qui traverse le circuit ?

2- Calculer les tensions efficaces U_L aux bornes de la bobine, U_R aux bornes de la résistance, U_C aux bornes du condensateur. Quel est le déphasage ? de l'intensité par rapport à la tension ?

3- Tracer le diagramme de Fresnel correspondant.

LES SYSTÈMES TRIPHASÉS

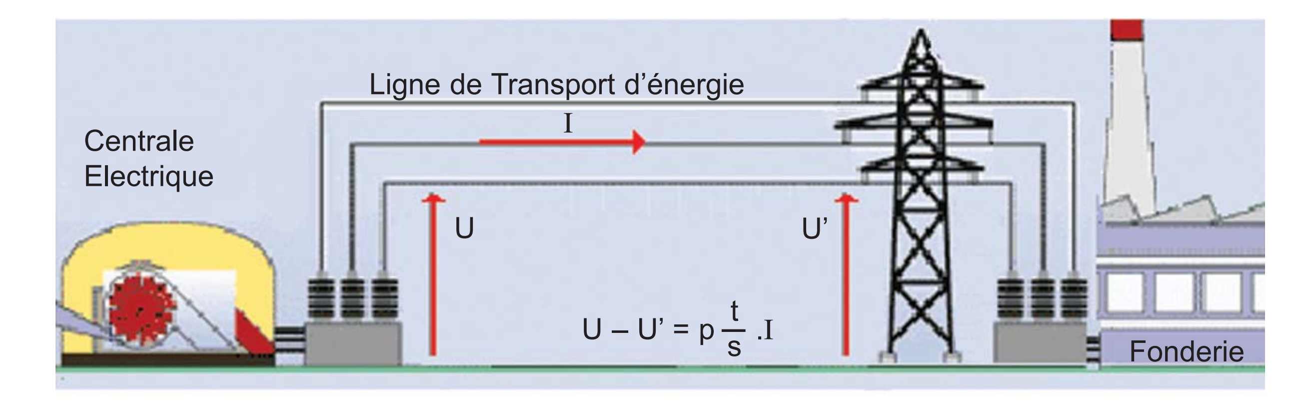
A- MISE EN SITUATION

1- Présentation du système :

Le haut fourneau



Au haut fourneau, première étape de la fabrication de tôles fortes, on produit de la fonte brute à partir du minerai de fer.
Dès cette phase, la constitution de la charge en minerais et additions est adaptée de manière précise au produit final, à savoir la tôle forte.
La demande d'électricité par cette industrie est devenue énorme ces dernières années.
L'appel de très fort courant surcharge les lignes de transport d'énergie et engendre des pertes en lignes considérables et des chutes de tension non négligeables. Il est alors devenu impératif d'utiliser des systèmes triphasés adaptés à un réseau de distribution trois fils dans le but de réduire les puissances perdues par effet joules dûes à la résistance du métal des fils conducteurs.



2- Problèmes posés :

a) Comment peut-on réduire les pertes en lignes ?
b) Quel système utiliser pour palier aux chutes de tension en lignes ?
c) Quels sont les précautions à prendre lors du branchement du système sur le réseau triphasé ?

B- AVANTAGES DU RESEAU TRIPHASÉ :

Activité
Réaliser l'activité N°1 du TP-B12 dans le manuel d'activités
Il s'agit d'accoupler une même charge :
- à un moteur asynchrone triphasé
- à un moteur asynchrone monophasé

de mesurer dans les deux cas le courant appelé par ligne et de comparer les résultats.

1- Avantages du triphasé par rapport au monophasé

- Les machines triphasées ont des puissances de plus de 50% supérieures aux machines monophasées de même masse et donc leurs prix sont moins élevés (le prix est directement proportionnel à la masse de la machine).
- Lors du transport de l'énergie électrique, les pertes sont moindres en triphasé qu'en monophasé.

2- Distribution de l'énergie électrique :

Chaque fois que l'on allume une lampe électrique ou que l'on démarre un moteur, il faut simultanément produire et transporter l'énergie. L'une des raisons principales du succès de l'électricité tient à ce qu'elle est très facilement transportable.

Un réseau triphasé permet d'alimenter des récepteurs à l'aide de trois conducteurs alors qu'il faudrait trois fois deux conducteurs (aller et retour) avec un réseau monophasé, ou deux conducteurs passant le triple du courant. L'économie sur la section de conducteur est évidente.

Un réseau triphasé est à priori plus économique qu'un réseau monophasé.

Remarque : Dans le réseau d'alimentation tunisien, le module des tensions simples est de 240 V. Il en résulte que celui des tensions composées est de 415 V.
Lorsqu'on caractérise un réseau triphasé par une seule tension, il s'agit toujours de la tension composée. On parle ainsi de réseau triphasé à 415 V.



Lignes de transport de l'énergie électrique

C- PROPRIETES DU RESEAU TRIPHASÉ :

Activité
Réaliser l'activité pratique N°2 du TP-B12 dans le manuel d'activités
Il s'agit, pour une alimentation triphasée, de verifier les relations entre tension simple et tension composée.

1- Systèmes triphasés équilibrés

a/ Définition :

Un système triphasé est dit équilibré lorsque les trois tensions (composées ou simples) possèdent la même valeur efficace et qu'elles sont déphasées de 2≠/3 l'une par rapport à l'autre.

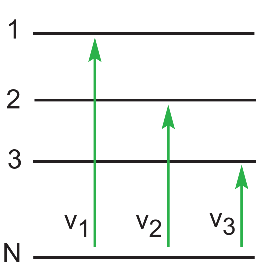
b/ Etude des tensions simples :

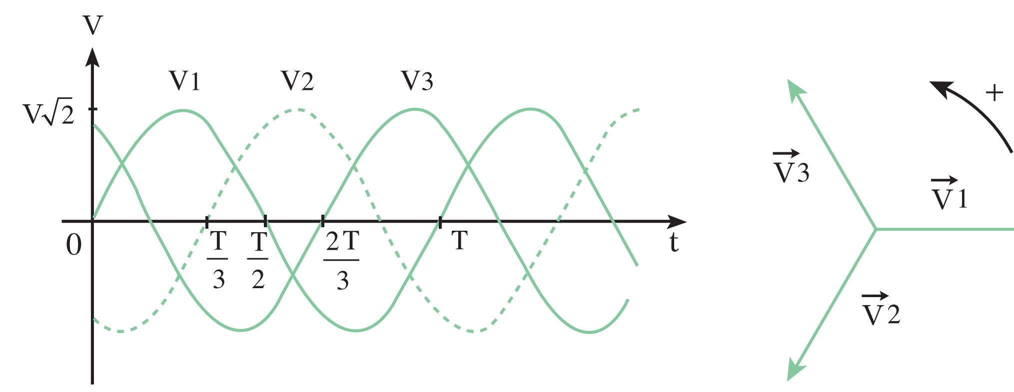
La tension simple est la différence de potentiel entre l'une des phases du réseau triphasé et la ligne neutre.

$$v_1(t) = V\sqrt{2}\ \sin(\omega t)$$

$$v_2(t) = V\sqrt{2}\ \sin(\omega t - \frac{2\pi}{3})$$

$$v_3(t) = V\sqrt{2}\ \sin(\omega t - \frac{4\pi}{3})$$

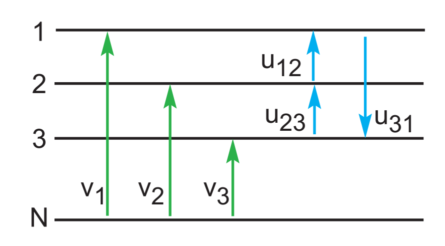




Oscillogrammes (représentation cartésienne) **Vecteurs de Fresnel associés (rep. vectorielle)**

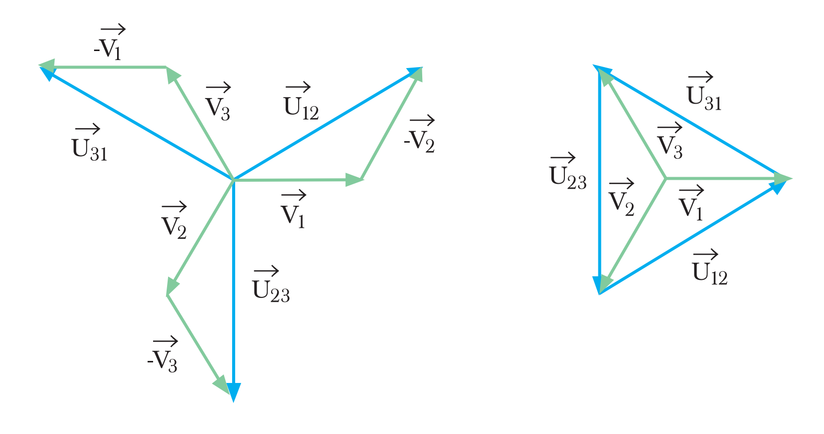
c/ Etude des tensions composées :

Les tensions composées ont la même fréquence que les tensions simples. Elles sont mesurées entre deux phases du réseau triphasé.
En application de la loi de Kirchhoff (loi des mailles) sur les tensions, les relations suivantes entre tensions simples et tensions composées peuvent être établies :



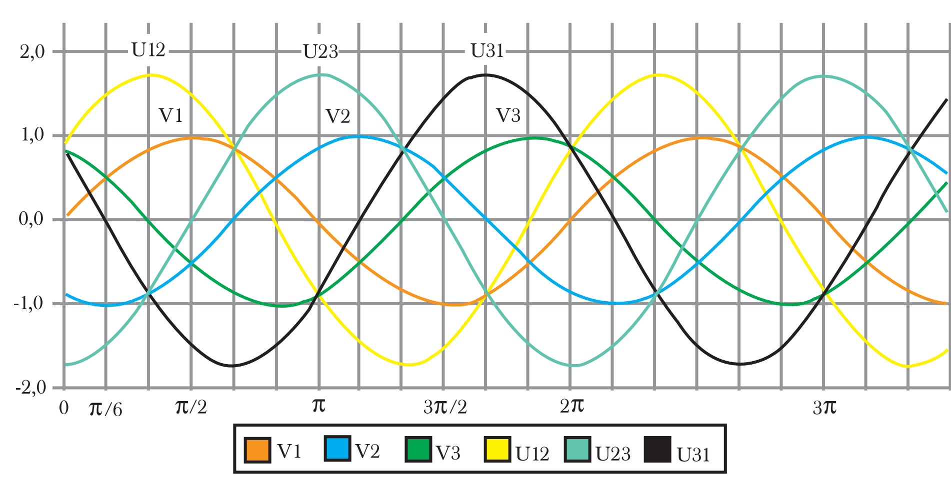
$$u_{12} = v_1 - v_2 \Rightarrow \vec{U}_{12} = \vec{V}_1 - \vec{V}_2$$
$$u_{23} = v_2 - v_3 \Rightarrow \vec{U}_{23} = \vec{V}_2 - \vec{V}_3$$
$$u_{31} = v_3 - v_1 \Rightarrow \vec{U}_{31} = \vec{V}_3 - \vec{V}_1$$

Les tensions composées forment donc également un système triphasé symétrique en avance de $\pi/6$ par rapport aux tensions simples.
Le diagramme vectoriel est le suivant :



Les équations horaires des tensions composées et leur oscillogrammes :

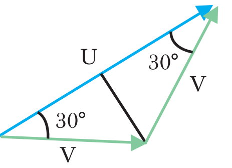
$$u_{12}(t) = U\sqrt{2}\sin(\omega t + \frac{\pi}{6} \;;\; u_{23}(t) = U\sqrt{2}\sin(\omega t - \frac{\pi}{2}) \;;\; u_{31}(t) = U\sqrt{2}\sin(\omega t - \frac{7\pi}{6})$$



d/ Relation entre tensions composées et tensions simples :

$$U = 2.V.\cos 30 \Rightarrow U = 2.V.\frac{\sqrt{3}}{2}$$

$$U = V.\sqrt{3}$$



Cette relation est toujours vraie quelque soit la charge.

D- RECEPTEUR TRIPHASÉ :

Activité
Réaliser l'activité N°3 du TP-B12 dans le manuel d'activités
Il s'agit, pour une charge triphasée couplée en étoile puis en triangle, de mesurer dans chaque cas :
- le courant dans chaque élément de la charge
- le courant de chaque ligne d'alimentation
- la tension aux bornes de chaque élément de la charge
- la tension entre deux phases de l'alimentation.

1- Introduction

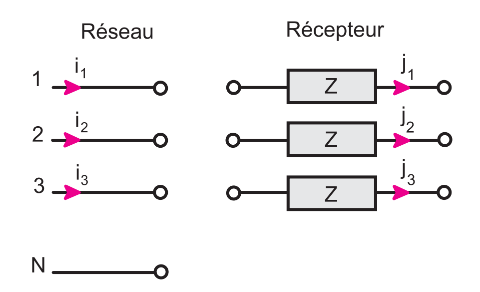
Des fours à céramique pour des artisans potiers, jusqu'aux fours à haute température pour la céramique technique, une large gamme de fours électriques est nécessaire. Pour réduire les courants de ligne, les résistors de chauffage sont alimentées par un réseau triphasé. Un tel système de chauffage constitue une charge triphasée généralement équilibrée.


Céramique

Tmax °C	Volume en L	Dimensions extérieures en mm			Puissance/ kM	Tension de branchement
		larg	prof	haut		
1200	20 30	900 1100	1200 1200	1350 1350	9 12	Triphasé

2- Définition

Un **récepteur triphasé équilibré** est un récepteur constitué de **trois éléments identiques**, d'impédance **Z**. Le courant par phase, noté **J** est celui qui traverse les éléments Z du récepteur triphasé.
Le courant en ligne, noté **I** est celui dans les fils du réseau triphasé.

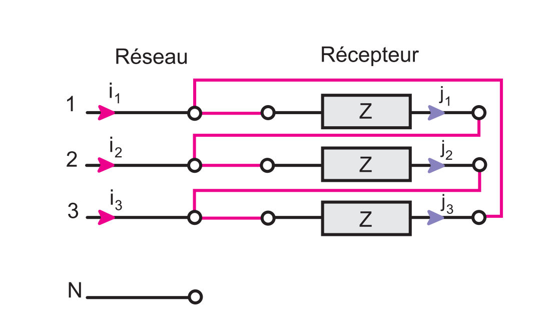




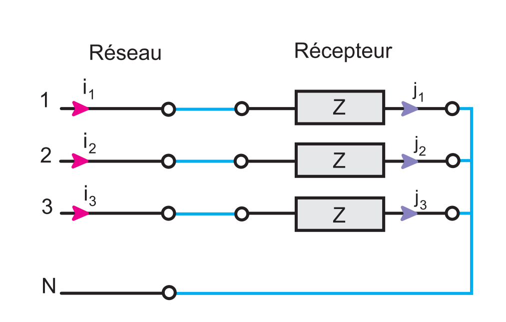
3- Branchement d'un récepteur triphasé :

Le réseau et le récepteur peuvent se relier de deux façons différentes suivant la tension qu'on applique à chaque élément :

- chaque élément est branché entre deux phases. Il reçoit la tension composée. Ce mode de couplage est dit **triangle**.



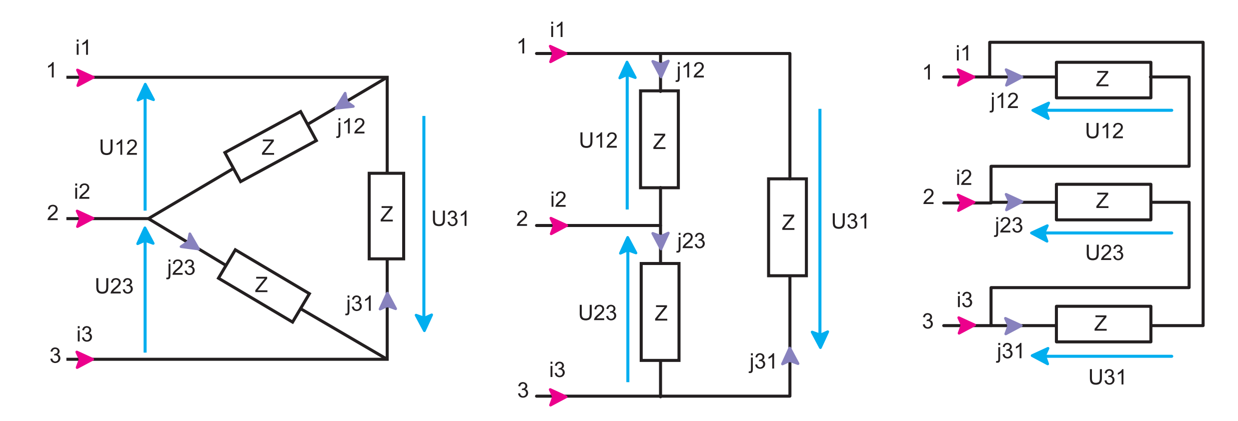
- chaque élément est branché entre phase et neutre. Il reçoit la tension simple. Ce mode de couplage est dit **étoile**.



4- Branchement en triangle d'un récepteur triphasé :

En appliquant le principe adopté ci-dessus sur les trois éléments d'un récepteur triphasé, on a trois façons différentes de représenter un couplage triangle.

NB : Le premier schéma explique le terme « triangle ».



D'après la loi des noeuds :

$$i_1 + i_2 + i_3 = 0 \quad \text{et} \quad j_{12} + j_{23} + j_{31} = 0$$

- Relations entre les tensions

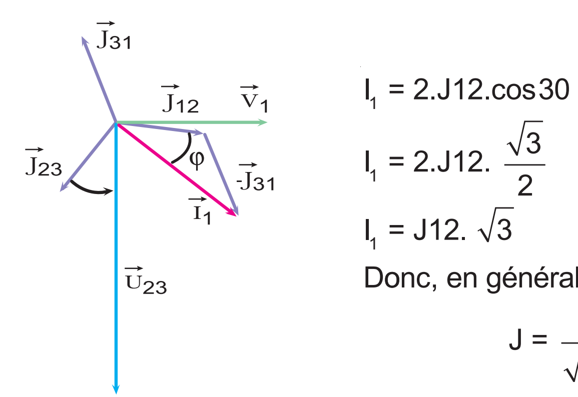
$$v_{z1} = v_{z2} = v_{z3} = U$$

- Relations entre les courants

$$i_1 = j_{12} - j_{31} \Rightarrow I_{1max} = J_{12max} - J_{31max}$$
$$i_2 = j_{23} - j_{12} \Rightarrow I_{2max} = J_{23max} - J_{12max}$$
$$i_3 = j_{31} - j_{23} \Rightarrow I_{3max} = J_{31max} - J_{23max}$$

Le système triphasé est dit équilibré si :

$$J_{12} = J_{23} = J_{31} = J \quad \text{et} \quad \varphi_1 = \varphi_2 = \varphi_3 = \varphi \quad \text{alors} \quad I_1 = I_2 = I_3 = I$$



$I_1 = 2.J12.\cos 30$

$I_1 = 2.J12.\dfrac{\sqrt{3}}{2}$

$I_1 = J12.\sqrt{3}$

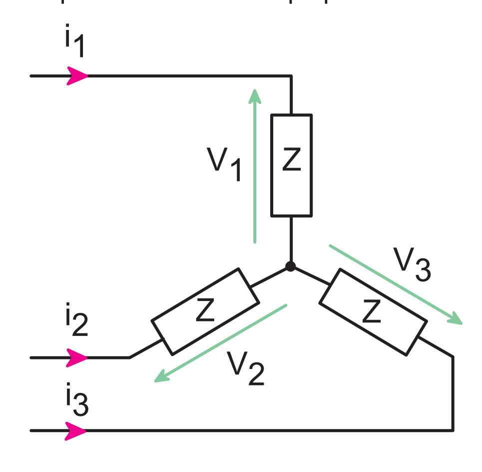
Donc, en généralisant, on peut écrire :

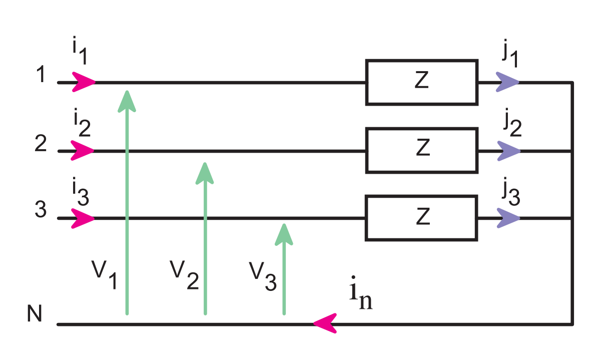
$$J = \frac{I}{\sqrt{3}}$$

5- Branchement en étoile d'un récepteur triphasé :

En appliquant le même principe sur les trois éléments d'un récepteur triphasé, on a deux façons différentes de représenter un couplage étoile.

NB : Le premier schéma explique le terme « étoile ».





Comme il s'agit des mêmes impédances, de ce fait

$$i_1 + i_2 + i_3 = 0 \ \text{donc} \ i_n = 0$$

Le courant dans le fil neutre est nul. Le fil neutre n'est donc pas nécessaire.
Pour un système triphasé équilibré, le fil neutre ne sert à rien.

- Relations entre les courants

On constate que les courants en ligne sont égaux aux courants par phase.

$$i_1 = j_1 \Rightarrow I_{1max} = J_{1max}$$
$$i_2 = j_2 \Rightarrow I_{2max} = J_{2max}$$
$$i_3 = j_3 \Rightarrow I_{3max} = J_{3max}$$

Le système triphasé est équilibré : $I_1 = I_2 = I_3 = I = J_1 = J_2 = J_3 = J$.

- Relations entre les tensions :

$$U_{12} = 2.V_{i1}.\cos 30$$

$$U_{12} = 2.V_{i1}.\frac{\sqrt{3}}{2}$$

$$U_{12} = V_{i1}.\sqrt{3}$$

Donc, en généralisant, on peut écrire : $V = \frac{U}{\sqrt{3}}$

D- MESURE DES PUISSANCES EN TRIPHASE :

Activité

Réaliser l'activité N°4 du TP-B12 dans le manuel d'activités

Il s'agit, pour une charge triphasée, de mesurer la puissance active consommée par la charge :

- par la méthode d'un seul wattmètre
- par la méthode des deux wattmètres

et d'en déduire :

- la puissance réactive consommée par la charge
- la puissance apparente
- le facteur de puissance.

1- Théorème de Boucherot :

Les puissances active et réactive absorbées par un groupement de dipôles sont respectivement égales à la somme des puissances actives et réactives absorbées par chaque élément du groupement.

donc d'après ce théorème :	P = P1+P2+P3 et Q = Q1+Q2+Q3
pour un récepteur équilibré :	P1=P2=P3 et Q1=Q2=Q3
finalement :	P=3.P1 et Q=3.Q1
facteur de puissance :	$\cos \varphi = P / S$.

2- Mesure de la puissance :

Suivant la nature de la charge, deux procédés de mesure sont utilisés :

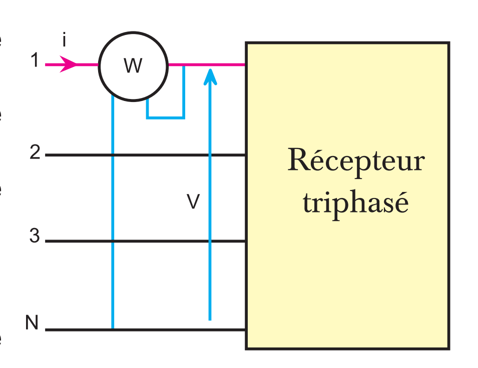
- dans le cas où le fil neutre est accessible, on utilise un seul wattmètre pour mesurer la puissance active totale.

Le wattmètre branché de cette façon mesure (puissance lue) : $P' = V.\ I.\ \cos \varphi$

La puissance du récepteur s'exprime (puissance absorbée) :

$$P = \sqrt{3}.\ U.\ I.\ \cos\varphi$$

La relation entre la puissance lue et la puissance absorbée par le récepteur est donc' : **P = 3 . P'**





NB : ✔ Ce procédé n'est valable que lorsque la charge est équilibrée
✔ Il n'est pas nécessaire de connaître le couplage du récepteur.

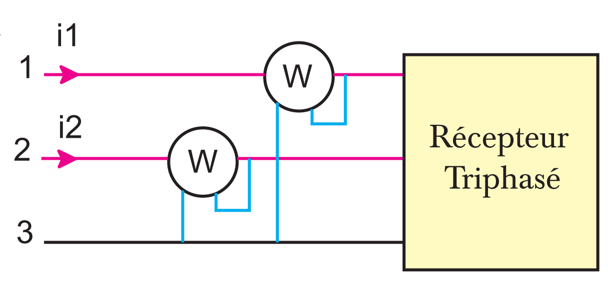
- dans le cas où le fil neutre n'est pas accessible, on utilise deux wattmètres pour mesurer la puissance active totale.

Le premier wattmètre mesure :

$P_1 = U_{13} . I_1 . \cos(\varphi - \pi / 6)$

Le deuxième wattmètre mesure :

$P_2 = U_{23} . I_2 . \cos(\varphi + \pi / 6)$



La puissance active totale absorbée par le récepteur est la somme algébrique des deux puissances mesurées P_1 et P_2 :

$$P = P_1 + P_2$$

NB : a/ la somme algébrique veut dire que si la puissance mesurée par l'un des deux wattmètres branché comme l'indique le schéma est négative, on inverse sa polarité en courant ou (exclusive) en tension et la puissance totale est alors la différence entre les deux puissances mesurées.

b/ a puissance réactive absorbée par le récepteur est donnée par : $Q = \sqrt{3} .(P_1 - P_2)$

3- Application : pertes par effet joules dans un récepteur triphasé

- dans le cas d'un couplage triangle

Ne considérons que la partie résistive du récepteur.
Détail du calcul de la résistance équivalente vue entre deux bornes du récepteur :
nous avons 2r en parallèle avec r ; $R_a = \dfrac{2.r.r}{2.r + r} = \dfrac{2}{3}.r$

Pour une phase du récepteur : $P_{j1} = r.J^2$

Résistance vue entre deux bornes : $R_a = \dfrac{2}{3}.r$

Pour le récepteur complet : $P = 3.P_{j1} = 3.r.J^2 = 3.\dfrac{3}{2}.Ra.(\dfrac{I}{\sqrt{3}})^2 = \dfrac{3}{2}.Ra.I^2$

Finalement pour le couplage étoile : $P = \dfrac{3}{2}.Ra.I^2$

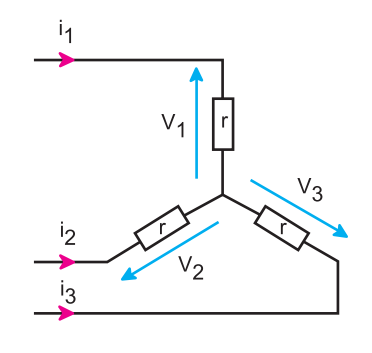
- dans le cas d'un couplage étoile

Ne considérons que la partie résistive du récepteur.
Pour une phase du récepteur : $P_{j1} = r.I^2$

Résistance vue entre deux bornes : $R_a = 2r$

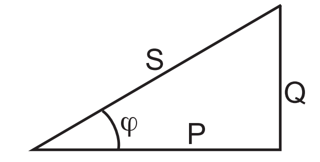
Pour le récepteur complet : $P = 3.P_{j1} = 3.r.I^2 = \dfrac{3}{2}.Ra.I^2$

Finalement pour le couplage étoile : $P = \dfrac{3}{2}.Ra.I^2$

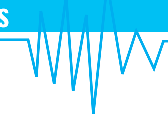


E- RESUMÉ

	Couplage étoile	Couplage triangle
Relation entre U et V	$U = V\sqrt{3}$	$U = V$
Relation entre I et J	$I = J$	$I = J\sqrt{3}$
Déphasage	φ (I, V)	φ (J, U)
Puissance active	$P = 3.P_1 = 3VI\cos\varphi$ $P = \sqrt{3}UI\cos\varphi$	$P = 3.P_1 = 3UJ\cos\varphi$ $P = \sqrt{3}UI\cos\varphi$
Pertes joules	$P = 3rI^2$ $P = \frac{3}{2}RI^2$	$P = 3J^2$ $P = \frac{3}{2}RI^2$
Résistance équivalente	$R = 2r$	$R = \frac{2}{3}r$
Puissance réactive	$Q = \sqrt{3}\,UI\sin\varphi$	$Q = \sqrt{3}\,UI\sin\varphi$
Puissance apparente	$S = \sqrt{3}UI$	$S = \sqrt{3}UI$
Facteur de puissance	$\cos\varphi = \frac{P}{S}$	$\cos\varphi = \frac{P}{S}$

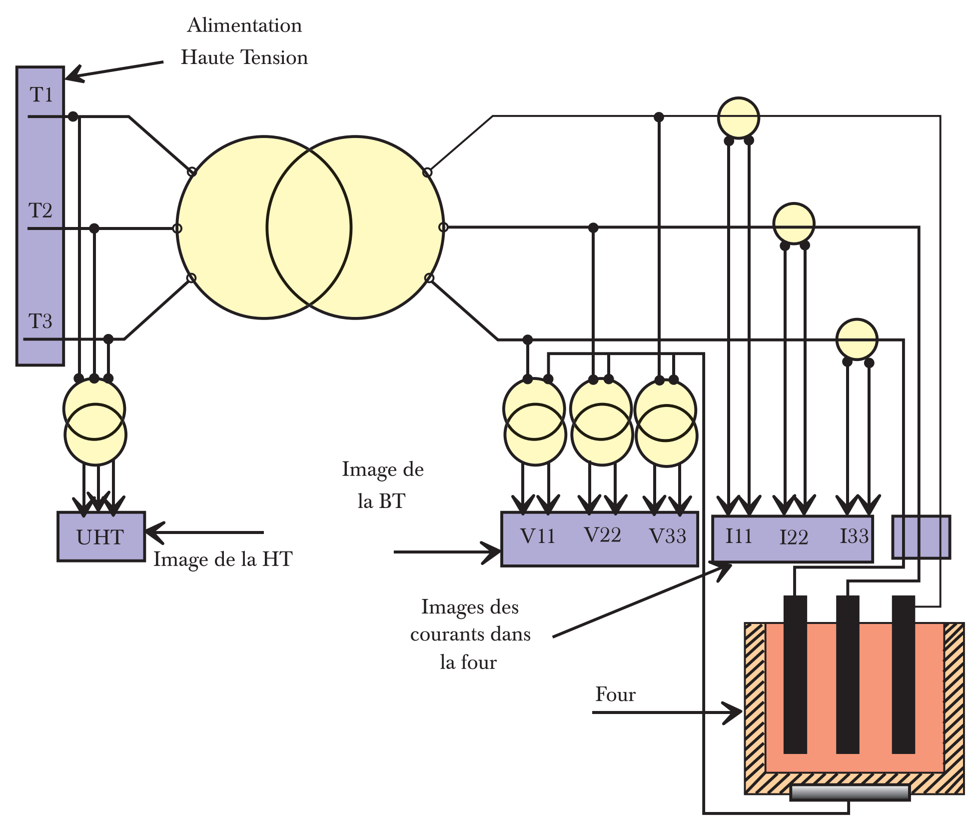


$$S = \sqrt{P^2 + Q^2}$$



Système d'étude : Le Haut Fourneau

Les courants d'arc, de l'ordre de 50 000 ampères, sont obtenus en imposant entre chacune des trois électrodes et la masse à fondre dans le four une tension.
L'énergie électrique est fournie par le réseau triphasé dans le but de limiter au maximum les pertes en ligne et des chutes de tension importantes.



F- EXERCICES D'APPLICATION :

1- Exercice résolu :

Un réseau électrique triphasé 220/380V, 50Hz, alimente une installation électrique comprenant :

- 60 lampes de 500W chacune (cosφ =1 à 220V), réparties de façon à équilibrer les trois phases;
- un groupe moteur aux bornes duquel la méthode des deux wattmètres a donné les indications suivantes :

  P1 = 200 kW   P2 = 60 kW ;
- **un four thermique 220/380V absorbant une puissance 1500 W ;**

1) Quelles sont les puissances active, réactive et apparente absorbées par le groupe moteur
2) Quels sont l'intensité du courant et le facteur de puissance à l'entrée de la dérivation du groupe moteur.
3) Quels sont l'intensité du courant et le facteur de puissance en tête de réseau.

Corrigé :

1) Puissances moteurs :

$$P_m = P_1 + P_2 - 200 + 60 = 260kW$$

$$Q_m = \sqrt{3}\,(P_1 - P_2) = 242,48\ kVAR$$

$$S_m = \sqrt{P_m^2 + Q_m^2} = 355,5\ kVA$$

2) Courant-facteur de puissance

$$I_m = \frac{S_m}{\sqrt{3}.U} = 540A \qquad \cos\varphi = \frac{P_m}{S_m} = 0,73$$

3) Courant et facteur de puissance totaux :

$$P_t = P_m + P_l + P_f = 291,5\ kW \qquad Q_t = Q_m = 242,48\ kVAR$$

$$S_t = \sqrt{P_m^2 + Q_m^2} = 372\ kVA$$

$$I_t = \frac{S_t}{\sqrt{3}U} = 576A \qquad \cos\varphi\ \frac{P_t}{S_t} = 0,768$$

2- Exercice à résoudre :

Exercice 1 :

Une installation alimentée par le secteur 220/380 V, 50 Hz comprend :
- un moteur triphasé (M1) : Pa = 5 kW, cos φ_1 = 0,7;
- un moteur triphasé (M2) : Pa = 8 kW, cos φ_2 = 0,8.

1) Calculer les courants dans les fils de ligne quand les deux moteurs fonctionnent en même temps.
2) Calculer le facteur de puissance de l'ensemble.
3) Déterminer les indications de chaque wattmètre suivant la méthode des deux wattmètres.

Exercice 2 :

Une installation électrique sans neutre alimentée par le réseau triphasé U = 380 V, 50 Hz comprend :
- 3 moteurs triphasés de caractéristiques nominales : P_M = 10 kW Q_M = 3kVAR
- 3 impédances Z montées en triangle : P_Z = 5 kW Q_Z = 3,2 kVAR (pour les trois impédances)
- 3 résistances montées en étoile : P_Z = 1 kW (par résistance).

1) Faire un schéma de l'installation.
2) Quelles sont les intensités dans une phase de chacun des circuits ?
3) Quel est le courant dans l'installation lorsque tous les appareils fonctionnent ?
4) Quel est le facteur de puissance global de l'installation ?
5) On mesure les puissances active et réactive fournies par l'installation au réseau en utilisant la méthode des deux wattmètres. Quelles seront les indications relevées sur les appareils de mesure ?

LES TRANSFORMATEURS MONOPHASÉS

A- MISE EN SITUATION

1- Présentation du système : Appareils de chauffage par induction
(montage à chaud)



Le montage à chaud permet de dilater le roulement et de le monter sans effort sur son arbre. Par rapport au bain d'huile, à la table chauffante ou aux étuves, les appareils de chauffage par induction sont plus sûrs et plus rationnels.

L'industrie a récemment développé une nouvelle gamme d'appareils de chauffage par induction avec des produits différents pour permettre le montage sans effort des roulements.

La technologie du chauffage par induction est propre et offre de multiples avantages : sécurité de l'opérateur, démagnétisation du roulement, facilité de maintenance ou encore choix du mode de fonctionnement (mode température / temps).
Le contrôle de la température est assuré par une sonde intégrée qui permet d'éviter :
- le risque de dépassement de la température affichée ou d'annulation du jeu du roulement,
- le risque de surchauffe des pièces (par défaut, l'appareil sélectionne une température de 110°C, température de chauffage recommandée pour les roulements).

2- Problèmes posés :

a) Comment s'est déroulée cette transformation d'énergie électrique/calorifique ?
b) Que représente le roulement dans ce système?
c) Quel est le rôle de l'armature carrée ?

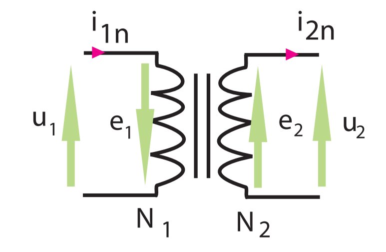
B- PRINCIPE DE FONCTIONEMENT D'UN TRANSFORMATEUR

Activité
Réaliser l'activité pratique N°1 du TP-B2 dans le manuel d'activités
Il s'agit de placer deux bobines de N1 et N2 spires autour d'un cicuit magnétique ouvert puis fermé et de mesurer dans les deux cas la fem induite aux bornes de la deuxième bobine.

$e_1 = -N_1 \frac{d\varphi}{dt}$ et $u_1 = -e_1$

U_1 est sinusoïdale

Posons : $u_1 = U_1\sqrt{2}\cos\omega t$



1- Création d'une force électromotrice induite :

L'enroulement du primaire, formé de N1 spires, est le siège d'une f.é.m. :

D'où : $\frac{d\varphi}{dt} = \frac{U_1\sqrt{2}}{N_1} \cos \omega t$

Et : $\varphi = \frac{U_1\sqrt{2}}{N_1\omega} \cos (\omega t - \frac{\pi}{2})$

Soit : $B \frac{U_1\sqrt{2}}{N_1 s\omega} \cos (\omega t - \frac{\pi}{2})$

Posons : $B_{max} = \frac{U_1\sqrt{2}}{N_1 s\omega}$ la valeur maximum du champ magnétique $\vec{B}$

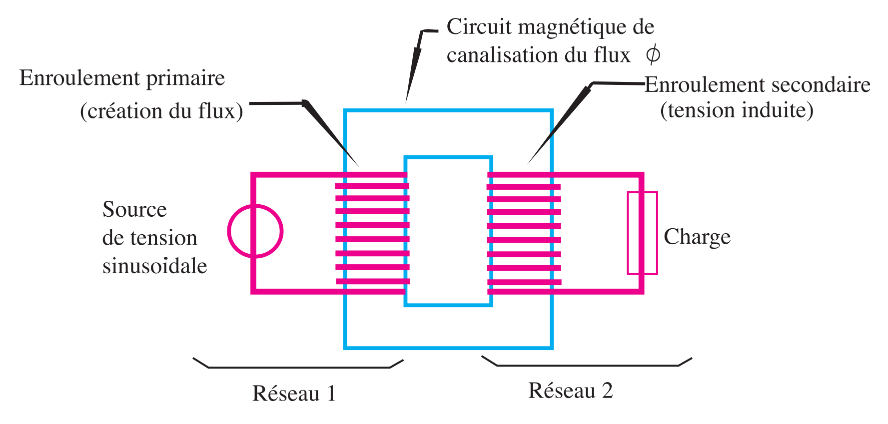
Comme $\omega t = 2\pi f$, nous obtenons l'expression de la valeur efficace U_1 :

$U_1 = 4,44\ N_1 fs\ B_{max}$ c'est la formule de Boucherot.

Le flux étant défini par la relation **φ = B.S** où **B** est le module de l'induction **B** du champ magnétique qui traverse perpendiculairement les spires et **S** la section (supposée constantes) de ces spires.

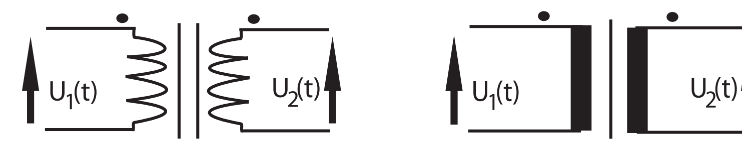
2- Constitution d'un transformateur :

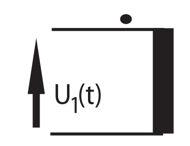
Le transformateur est une machine électrique statique permettant de transférer l'énergie électrique en adaptant les niveaux de tension (de nature sinusoïdale) et de courant entre deux réseaux de même fréquence.
Il est constitué de deux parties électriques isolées, l'enroulement primaire et l'enroulement secondaire, liées magnétiquement par un circuit magnétique (Figure ci-dessous).
Pour des impératifs de fabrication et d'efficacité la réalisation pratique donne à ce dernier une autre forme : circuit magnétique cuirassé et bobinages concentriques.

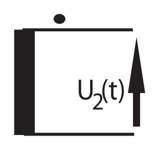


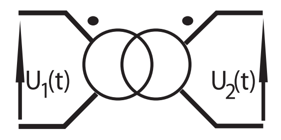
Constitution du transformateur

3- Symboles utilisés pour représenter un transformateur :









4- Relation entre les tensions

Pour un transformateur parfait tout le flux créé par l'enroulement primaire traverse l'enroulement secondaire.

$$u_1 = -e = N_1 \frac{d\varphi}{dt} \quad \text{et} \quad u_2 = e_2 - N_2 \frac{d\varphi}{dt}$$

Soit : $\frac{u_2}{u_1} = -\frac{N_2}{N_1} = -m$

ou encore : $u_2 = -mu_1$

m est le rapport de transformation. Les tensions $\mathbf{u_1}$ et $\mathbf{u_2}$ sont **en opposition de phase.**

B- FONCTIONNEMENT A VIDE D'UN TRANSFORMATEUR :

Activité

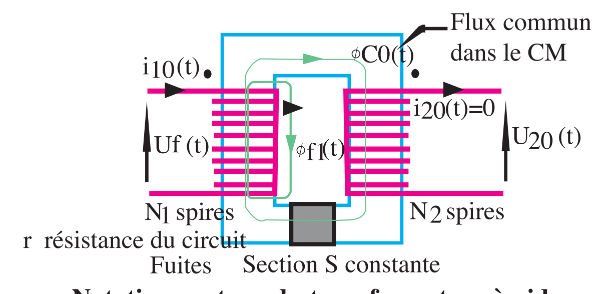
Réaliser l'activité pratique N°2 du TP-B2 dans le manuel d'activités

Il s'agit de faire fonctionner un transformateur monophasé à vide et de mesurer :

- les tensions primaire et secondaire d'un transformateur monophasé et d'en déduire une relation entre ces deux tensions
- la puissance absorbée par le transformateur monophasé à vide et d'en déduire les pertes fer et $\mathbf{\cos\varphi_{10}}$ (On désigne par $\mathbf{\cos\varphi_{10}}$ le déphasage entre courant et tension au primaire et à vide)

1- Conditions de fonctionnement :

Le transformateur comporte deux enroulements de résistances $\mathbf{r_1}$ et $\mathbf{r_2}$ comportant $\mathbf{N_1}$ et $\mathbf{N_2}$ spires (Figure ci-contre). Le primaire reçoit la tension $\mathbf{u_1(t)}$ et absorbe le courant $\mathbf{i_{10}(t)}$. Le secondaire délivre la tension $\mathbf{u_{20}(t)}$ et un courant $\mathbf{i_{20}(t)}$ nul puisqu'il est à vide.

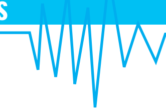


Notations autour du transformateur à vide

Le rapport de transformation se mesure à vide (pas de charge, I_2=0)

$$m = \frac{U_{20}}{U_1} = \frac{N_2}{N_1}$$

si **m > 1**, le transformateur est **élévateur** de tension ;
si **m < 1**, le transformateur est **abaisseur** de tension.



2- Puissance consommée à vide :

A vide, le transformateur absorbe une puissance active P_{10} et une puissance réactive Q_{10}.
On peut écrire : $\mathbf{P_{10} = U_1\ I_{10}\ cos\varphi_{10}}$ où $cos\varphi_{10}$ est le facteur de puissance à vide.
$\mathbf{Q_{10} = U_1\ I_{10}\ sin\varphi_{10}}$
Si on considère que le flux de fuite est nul :
-toute la puissance réactive absorbée est transformée en puissance magnétisante.
-toute la puissance active absorbée par le transformateur est dissipée en chaleur :
- ✔ par effet joules dans les bobinages du primaire.
- ✔ par échauffement de la masse métallique du circuit magnétique (pertes fer)

> $P_{10} = P_{J10} + P_{fer}$
> $I_{10} << I_{10}$ donc $P_{J10} = r_1 I_{10}^2 << P_{10}$
> Les pertes fer dépendent de la fréquence du réseaux. Elles sont constantes pour une fréquence donnée.
> $P_{J10} << P_{fer}$. Donc $P_{10} = P_{fer}$.

C- FONCTIONNEMENT EN CHARGE D'UN TRANSFORMATEUR :

Activité
Réaliser l'activité pratique N°3 du TP-B2 dans le manuel d'activités
Il s'agit de réaliser un essai en charge où $\mathbf{U_2}$ et $\mathbf{cos\varphi_2}$ sont maintenus constants en vue de :
- ✔ mesurer les puissances mises en jeu au primaire et au secondaire d'un transformateur monophasé de faible puissance.
- ✔ tracer la caractéristique en charge $\mathbf{U_2 = f(I_2)}$ et d'en déduire la courbe de chute de tension $\mathbf{\Delta U_2 = f(I_2)}$.

1- Conditions de fonctionnement :

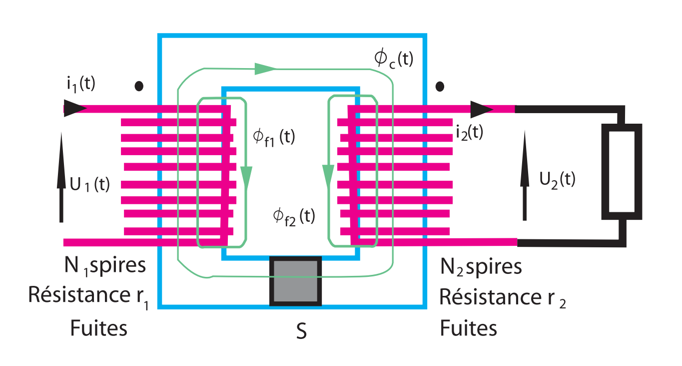
Le secondaire du transformateur est fermé sur une charge qui appelle un courant secondaire I_2.

On constate une chute de tension aux bornes du secondaire qui dépend de la valeur de I_2.

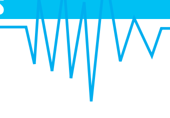
$$\mathbf{U_2 < m.U_1}$$

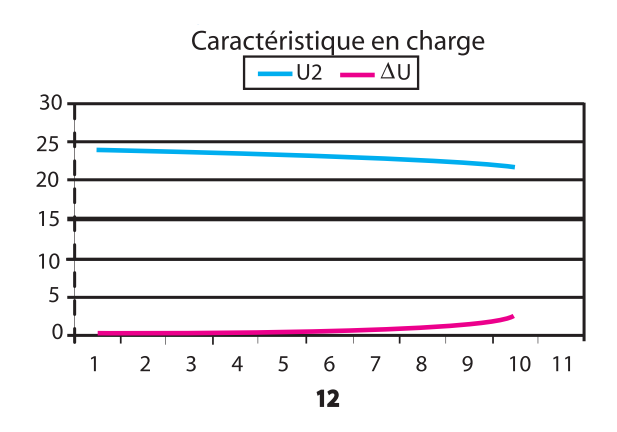
Cette dernière observation est le résultat d'une chute de tension provoquée par la résistance du bobinage secondaire

$$\mathbf{\Delta U = r_2.I_2}$$



(si I_2 augmente ΔU augmente aussi).





2- Puissance consommée en charge :

$P_1 = U_1 I_1 \cos\varphi_1$ où $\cos\varphi_1$ est le facteur de puissance en charge.

En charge, le transformateur absorbe une puissance active importante suffisante pour satisfaire :

- la puissance demandée par la charge :

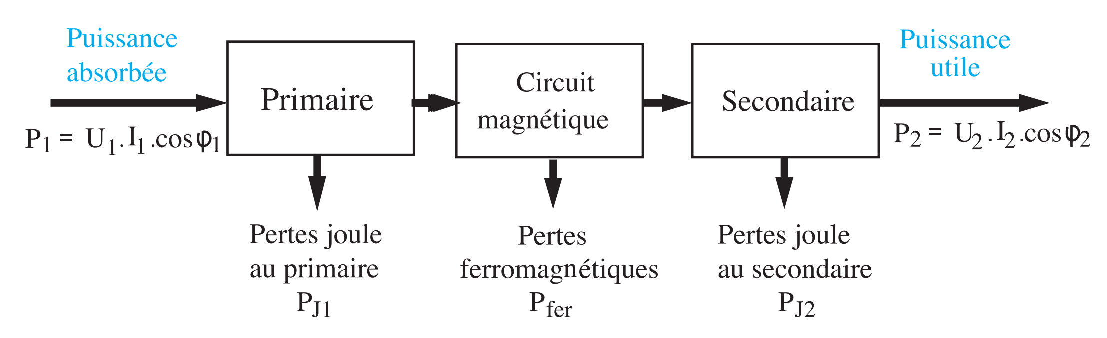
  $P_2 = U_2 I_2 \cos\varphi_2$ où $\cos\varphi_2$ est le facteur de puissance en charge
- les pertes joules primaire et secondaire (calculables) :

  $P_j = P_{j1} + P_{j2} = r_1.I_1^2 + r_2.I_2^2$
- les pertes fer (constantes, mesurables dans un essai à vide) :

  $P_f = P_{10}$

3- Rendement : $\eta = \frac{Pu}{Pa} = 1 - \frac{\Sigma\, pertes}{Pa}$

D- BILAN ENERGETIQUE ET RENDEMENT D'UN TRANSFORMATEUR:



$\eta = \frac{Pu}{Pa} = 1 - \frac{\sum pertes}{Pa}$	**Puissance absorbée**	$Pa = U_1 . I_1 . \cos\varphi_1$
	Pertes joules primaire	$P_{j1} = r_2 . I_1^2$
	Pertes joules secondaire	$P_{j2} = r_2 . I_2^2$
	Pertes fer	P_f est déterminée par un essai à vide
	Puissance utile	$P_u = P_a - \Sigma$ pertes $= P_a - (p_f + p_{j1} + p_{j2})$

Le rendement varie en fonction des conditions d'utilisation du transformateur. Le meilleur rendement s'obtiendra pour les grandeurs d'utilisation nominales indiquées sur la plaque signalétique du transformateur.
Les bons transformateurs de forte puissance peuvent atteindre un rendement de 98%.

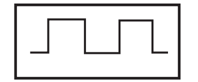
E- TRANSFORMATEURS SPECIAUX :

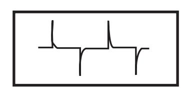
1- Transformateur d'isolement

Transformateur de rapport m = 1 utilisé pour l'isolation galvanique entre deux parties d'une installation électrique.
En effet, dans le transformateur, il n'y a aucun contact électrique entre le circuit primaire et le circuit secondaire.

2- Transformateur d'impulsion

Utilisé pour la commande de gâchette des thyristors, il transforme un signal carré en signal impulsionnel.





3- Transformateur d'intensité TI

Ce type de transformateur est particulier dans sa construction. Il est construit avec une bague bobinée qui représente le secondaire du transformateur, le primaire étant le fil ou la barre de cuivre qui passera à l'intérieur de l'anneau.
Le TI est principalement utilisé pour alimenter des compteurs d'énergies. Il ne fonctionne jamais à secondaire ouvert.

F- RESUMÉ :

Le transformateur est une machine électrique statique permettant de transférer l'énergie électrique en adaptant les niveaux de tension (de nature sinusoïdale) et de courant entre deux réseaux de même fréquence.

Fonctionnement à vide :

- Le rapport de transformation se mesure à vide (pas de charge, I_2=0) $m = \frac{U_{20}}{U_1} = \frac{N_2}{N_1}$
- La puissance consommée est transformée en chaleur par effet joule au primaire et par pertes fer dans le circuit magnétique $P_{10} = P_{j10} + P_{fer}$

Fonctionnement en charge :

- La puisance active consommée est transférée à la charge moyennant des pertes : $P_r = r_1 . I_1^2 + r_2 . I_2^2$
  - joules au primaire et au secondaire $P_{fer} = P_{10} + P_{j10}$
  - les pertes fer déterminées lors de l'essai à vide
- Le rendement est égal au rapport de la puissance utile à la puissance absorbée $\eta = \frac{Pu}{Pa} = 1 - \frac{\sum pertes}{Pa}$

Système d'étude : Appareil de chauffage par induction

C'est un transformateur dont le primaire est alimenté par une source de tension alternative de fréquence élevée et le secondaire est formé d'une seule spire fermée sur elle-même (courant secondaire très fort). Cette technologie offre l'avantage d'être propre, d'assurer la sécurité de l'opérateur et de permettre une maintenance facile.

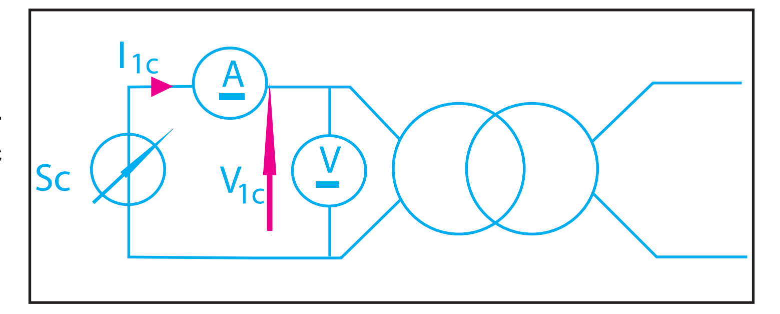
G- EXERCICES D'APPLICATION :

1- Exercice résolu :

Le primaire du transformateur étudié est alimenté par le réseau STEG sous une tension de valeur efficace V_{1N} = 225 V et de fréquence f = 50 Hz.

1/ Essai n°1

On a réalisé un essai en continu ; le schéma du montage est représenté sur la figure ci-contre. Sc désigne une source de tension continue réglable.



On a mesuré :

V_{1c} = 12 V ; I_{1c}= 3,64 A.

Calculer la valeur de la résistance R_1 du primaire.

2/ Essai n°2

Il s'agit d'un essai à vide réalisé sous tension primaire nominale, $V_{10} = V_{1N}$.

On a mesuré les grandeurs suivantes :

I_{10} = 0,24 A : valeur efficace de l'intensité du courant absorbé par le primaire.

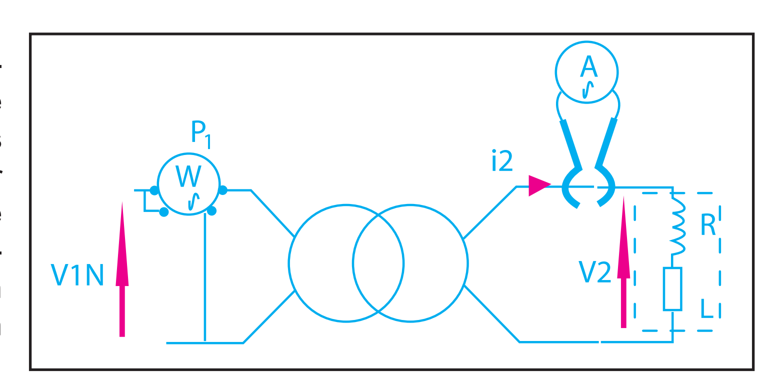
V_{20} = 48,2 V : valeur efficace de la tension secondaire à vide.

P_{10} = 10,2 W : puissance absorbée par le primaire.

- **a)** Calculer le rapport de transformation ou rapport du nombre de spires m = N_2/N_1
- **b)** Évaluer les pertes par effet Joule dans ce fonctionnement.
- **c)** En déduire la valeur des pertes dans le fer à vide et justifier l'emploi de cette même valeur en charge sous tension primaire nominale.

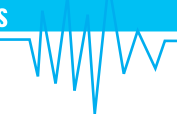
3/ Essai en charge nominale

Le schéma du montage est représenté sur la figure ci-contre; le transformateur est alimenté sous tension primaire nominale. Pour simuler la charge, on utilise une bobine sans noyau de fer, équivalente à un circuit RL série. Son impédance est Z = 11,6 Ω et son facteur de puissance cos = 0,89.



Le wattmètre mesure P_1 = 18 0 W et la pince ampèremétrique I_2 = 4,0 A.

- **a)** Calculer la tension secondaire en charge, V_2
- **b)** Montrer que la résistance R de la bobine est R = 10,3 Ω. En déduire la puissance active P_2 consommée par cette charge.
- **c)** Déterminer le rendement du transformateur au cours de cet essai.
- **d)** En déduire la valeur des pertes par effet Joule du transformateur.



Corrigé :

1/ Essai n°1

Cet essai en continu permet de mesurer la résistance R_1 de l'enroulement primaire. En effet, l'impédance du primaire se réduit en continu (f = 0) à la résistance R_1 (L_1.w=0).
D' après la loi d' Ohm : R1 = = 3.3 Ω

2/ Essai n°2 : à vide

a) m = = 0,214

b) p_{J0} = $R1\, I_{10}^2$ = 3.3 x $(0.24)^2$ = 0,19 W

c) Bilan de puissances à vide: $P_{10} = p_{j0} + p_{f0} \rightarrow p_{f0} = P_{10} - p_{j0}$ =10.2 - 0.19 =10 W

Les pertes dans le fer ne dépendent que de la tension U_1 et de la fréquence f : elles ne dépendent pas de la charge. L'essai à vide est réalisé sous tension primaire nominale : $p_{f0} = p_{fN}$.

3/ Essai en charge nominale

a) $V_2 = Z\, I_2$ = 11.6 x 4 = 46,4 V (Loi d' ohm en valeurs efficaces pour la charge).

b) Z = R + jLω = [Z ; φ] → R = Z cosφ = 11.6 x 0.89 = 10,3Ω

$P_2 = R\, I_2^2$ =10.3 x $(4)^2$ = 165 W : La puissance active est absorbée par la résistance R.

c) η= 92 %

d) Bilan de puissances : $P_2 = P_1 - p_j - p_{FN}$-> $p_j = P_1 - P_2 - p_{FN}$ = 180 - 165 - 10 = 5 W.

1- Exercice à résoudre :

Etude d'un transformateur monophasé :

Lors d'un essai à vide sous tension nominale du transformateur monophasé on a relevé :
U1 = 660 V ; f = 50 Hz ; I10 = 0,60 A ; U20 = 382 V ; P10 = 50 W.
Un courant continu d'intensité égale à 3,0 A correspond à une tension V1 = 6,3 V lorsqu'il circule au primaire et V2 = 2,1 V lorsqu'il circule au secondaire.

1- Calculer la résistance de chaque bobinage. Déterminer la résistance totale ramenée au secondaire RS. On rappelle que RS = m2 R1 + R2 (m : rapport de transformation).

2- Pour l'essai à vide, calculer les pertes par effet Joule. En déduire les pertes dans le fer. Déterminer l'erreur relative commise en assimilant la puissance absorbée à vide avec les pertes dans le fer.

3- Calculer le facteur de puissance du transformateur lors de l'essai à vide.

4- On note : I_{1r} la valeur efficace de la composante réactive de l'intensité i_{10} et I_{1a} la valeur efficace de la composante active ($i_{10} = i_{1a} + i_{1r}$: relation instantanée).
Calculer I_{1r} et I_{1a}.

LES MOTEURS ASYNCHRONES TRIPHASÉS

A- MISE EN SITUATION

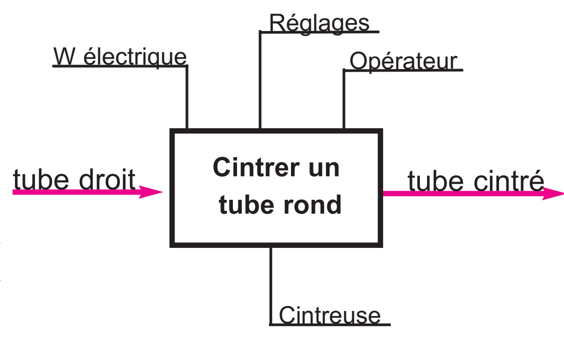
1- Présentation du système : Cintreuses industrielle de production

Alors que les tubes se cintrent encore au sable, l'industrie invente une machine permettant de cintrer les tubes à froid sans remplissage. La première cintreuse est née. Elle fonctionne par emboutissage avec un système de vérin à vis actionné par un cliquet.



En complément des outillages standards proposés pour les tubes les plus courants, les industriels étudient et réalisent à la demande des outillages spécifiques apportant des solutions adaptées à chaque problème de cintrage.

Le cintrage par enroulement permet de cintrer tous les tubes, avec des rayons plus courts que ceux utilisés pour le cintrage par emboutissage, et d'obtenir une meilleure qualité de cintrage. Il est particulièrement recommandé pour les tubes minces utilisés pour le mobilier et la serrurerie (tubes qualité 102, tubes suivant norme NFA 49-642) et les tubes inox 304, 316 etc...



Pour des dimensions importantes du tube à cintrer l'effort devient important et ne peut être fourni par des muscles humains. Il est alors indispensable de motoriser la machine en confiant le travail pénible à un vérin hydraulique alimenté par un compresseur entraîné par un **moteur électrique asynchrone.**

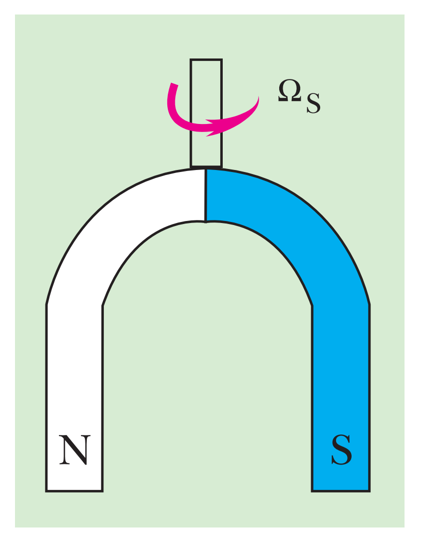
2- Problèmes posés :

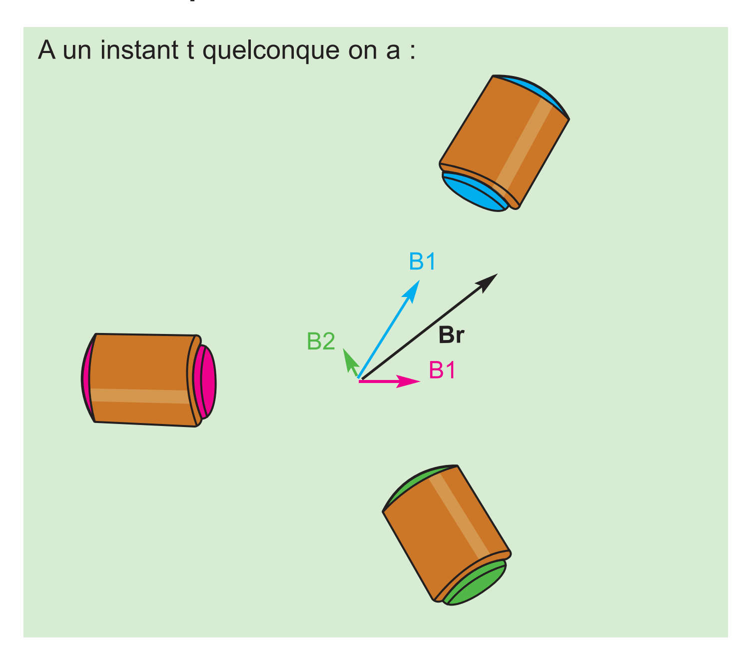
a) Qu'est ce qu'un moteur asynchrone ?
b) Quel moteur asynchrone peut-on choisir pour une application industrielle ?
c) Quel équipement doit on utiliser pour alimenter et protéger un moteur asynchrone dans son environnement de production ?

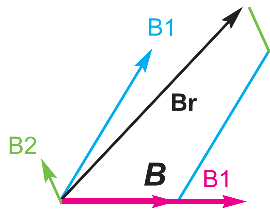
B- PRINCIPE DE FONCTIONNEMENT D'UN MOTEUR ASYNCHRÔNE

1- Phénomène de champ tournant :

L'aimant en forme de **U** en rotation autour d'un axe vertical avec ses deux pôles **NORD** et **SUD** en déplacement physique crée dans l'espace un **champ magnétique tournant**. Ce même phénomène peut être obtenu autrement en plaçant trois bobines physiquement fixes dans l'espace disposées à **120°** l'une par rapport à l'autre et alimentées par un **réseau alternatif triphasé**.







Br est un champ de module constant tournant à la vitesse angulaire Ω_S. Il est la résultante des 3 champs pulsants B1, B2 et B3 créés par les 3 bobines.

2- Principe de fonctionnement d'un moteur asynchrone :

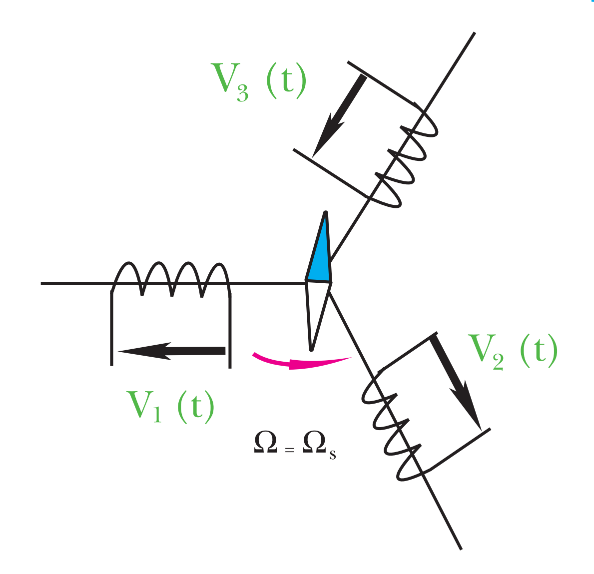
Activité
Réaliser l'activité pratique N°1 du TP-B3 dans le manuel d'activités
Il s'agit de comparer le fonctionnement :
- d'une aiguille aimantée
- d'un disque en cuivre ou en aluminium placés au milieu de trois bobines alimentées en courant alternatif.

Le principe du champ tournant créé par trois électro-aimants disposés à 120° est retenu car l'intensité de l'induction résultante **Br** dans ce cas peut être importante. Le décalage des électroaimants est déduit du déphasage relatif entre les trois tensions **V1**, **V2** et **V3** qui vaut $2\pi/3$.

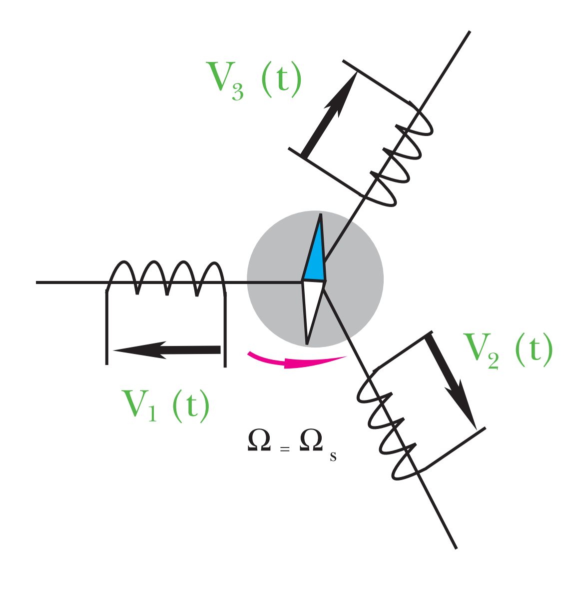
Les trois bobines sont fixes. Elles constituent le **stator** du moteur. La partie tournante doit pouvoir suivre le champ tournant et est appelée **rotor**. Elle peut être assimilée à un aimant placé dans l'axe géométrique de l'ensemble. Sa vitesse serait alors égale à la vitesse de rotation du champ magnétique. L'aiguille et le champ tournent à la même vitesse. On dit alors qu'il sont en **synchronisme**

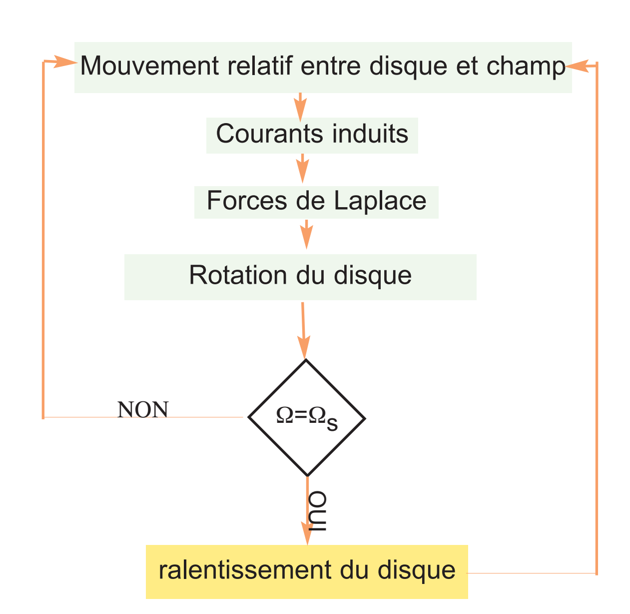
$$\omega_s = 2\pi f$$

avec f=50Hz pour le réseau STEG



Ce système n'est pas stable car si on soumet le rotor à un couple résistant, il devient incapable de suivre le champ tournant et par conséquent il décroche.
En remplaçant l'aimant par un disque métallique (aluminium, cuivre ou alliage léger) il est placé à l'arrêt dans un champ variable. Il est alors parcouru par des courants induits et sera donc soumis à un ensemble de **forces de Laplace** qui tenteront de le faire tourner dans le sens de rotation du champ magnétique.





C- CONSTITUTION D'UN MOTEUR ASYNCHRONE A ROTOR EN COURT-CIRCUIT

Activité

Réaliser l'activité N°2 du TP-B3 dans le manuel d'activités

Il s'agit d'identifier les différents éléments qui constituent un moteur asynchrône à rotor en court-circuit disponible au laboratoire :

1- Le stator (figure ci-dessous) :

Il est constitué de trois enroulements (bobines) parcourus par des courants alternatifs triphasés. Chaque bobine est divisée en ***p*** tranches réparties sur tout le stator et appelés **paires de pôles**. La vitesse de rotation du champ magnétique tournant dépend du nombre ***p*** de paires de pôles du stator.

$$\Omega_S = \frac{\omega}{p}$$

Ω_S : vitesse synchrone de rotation du champ tournant en rad.s^{-1}.

ω : pulsation des courants alternatifs en rad.s^{-1}.

$\omega = 2.\pi.f$

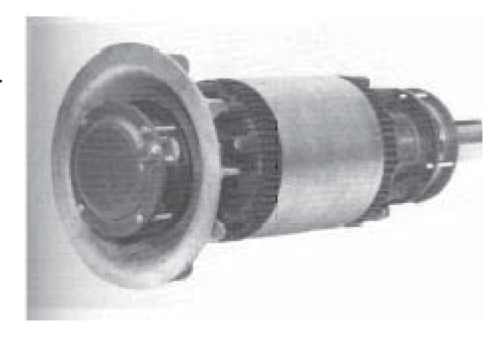
p : nombre de paires de pôles.



2- Le rotor (figure ci-contre) :

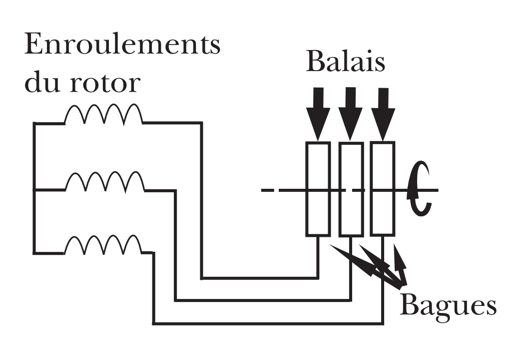
Le rotor est constitué d'un empilage de tôles minces en fer découpé pour créer des encoches.

Il n'est relié à aucune alimentation. Il peut être bobiné ou en cage d'écureuil :



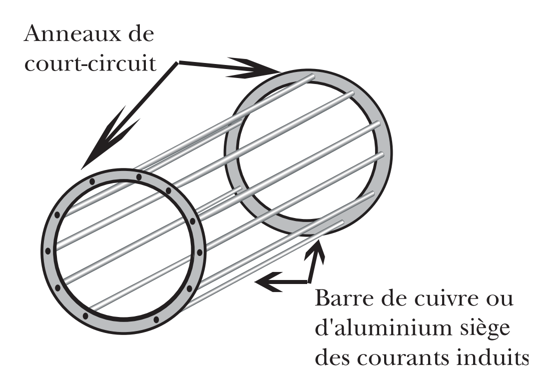
- rotor bobiné pour les machines de fortes puissances :

Les tôles de ce rotor sont munies d'encoches où sont logés des conducteurs formant des bobinages. On peut accéder à ces bobinages par l'intermédiaire de trois bagues et trois balais.



- rotor à cage d'écureuil pour les machines de moyenne ou faible puissance :

Il est constitué de barres conductrices très souvent en aluminium. Les extrémités de ces barres sont réunies par deux couronnes également conductrices.
On dit que le rotor est en court-circuit. Sa résistance électrique est très faible.



Le rotor tourne à une vitesse **Ω** plus petite que la vitesse de synchronisme Ω_S.
On dit que le rotor «glisse» par rapport au champ tournant.

Ce glissement **g** va dépendre de la charge (couple mécanique résistant).

$$g = \frac{\Omega_S - \Omega}{\Omega_S} = \frac{n_S - n}{n_S}$$

ns : vitesse de rotation de synchronisme du champ tournant ($tr.s^{-1}$).
n : vitesse de rotation du rotor ($trs.s^{-1}$).
$\Omega_S = 2\pi n_S$ ($rad.s^{-1}$) et $\Omega = 2\pi n$ ($rad.s^{-1}$)

Exemple :

Soit un réseau triphasé (f=50Hz) alimentant un moteur asynchrône à trois paires de pôles (p=3) :

n_S = 50/3 = 16,7trs/s = 1000 trs/min

A charge nominale ce moteur tourne à 950 trs/min :

g_n= (1000 - 950)/1000 = 0,05=5%

A vide (pas de charge), la vitesse n est voisine de ns :

$g_{vide} = 0$

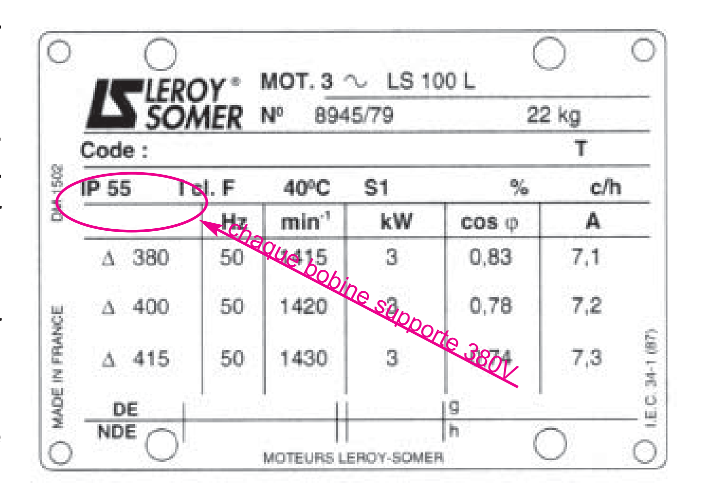
Au démarrage (n = 0) :

g = 1 = 100%

D- BRANCHEMENT D'UN MOTEUR ASYNCHRÔNE SUR LE RESEAU

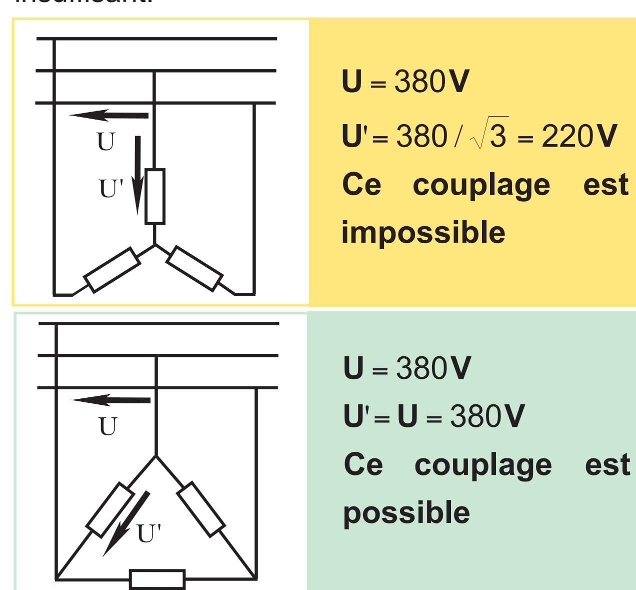
1- Couplage du stator :

Le problème consiste à adapter le bobinage du moteur à la nature du réseau sur lequel il sera branché.
La plaque signalétique poinçonnée visible sur le corps du moteur indique principalement la tension que peut supporter chaque bobine.



Exemple :
- sur une plaque signalétique d'un moteur on lit : 380 VΔ.
- le réseau est en 220V / 380 V.

De la plaque signalétique on déduit que la tension nominale d'une phase du moteur est de 380V.
Si on branche ce moteur en étoile, la tension au borne d'une phase sera de 220 V ce qui est insuffisant.

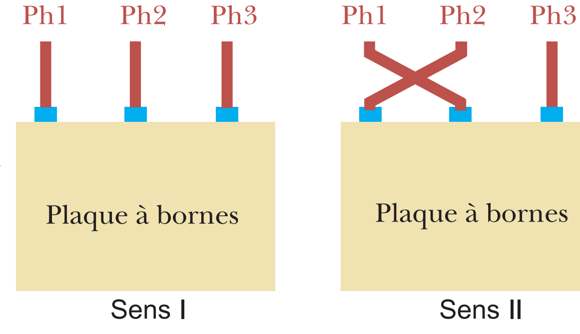


Conclusion :
ce moteur peut être branché uniquement en triangle sur le réseau 220/380V

Remarque :
Actuellement tous les moteurs supportent 380 V par phase. Ils supportent même souvent 400 V et 415 V, car le réseau STEG évolue progressivement vers ces tensions.

2- Inversion du sens de rotation :

Pour inverser le sens de rotation d'un moteur asynchrone triphasé, il suffit d'inverser deux fils d'alimentation du moteur.



Sens I
Sens II

E- FONCTIONNEMENT À VIDE

A vide le moteur n'entraîne pas de charge.
La puissance utile est nulle : **Pu = 0 W**
Le glissement est presque nul et le moteur tourne à la vitesse de synchronisme.

Activité
Réaliser l'activité N°3 du TP-B3 dans le manuel d'activités
Il s'agit de déterminer pour un moteur asynchrone tournant à vide :
- le courant à vide
- les pertes constantes
- la vitesse de rotation et le glissement.

A vide : $g \approx 0$ **et donc** $n \approx n_S$

Le facteur de puissance à vide est très faible (**cosφ < 0,2**) et le courant absorbée reste fort (**P** est petite et **Q** est grande). On parle alors de courant réactif ou magnétisant (ils sert à créer le champ magnétique).
La puissance active absorbée est faible. Elle est transformée totalement en chaleur :

- dans le stator à cause des pertes par effet joule : si **I** est le courant par phase et **R** sa résistance alors la somme des pertes joules dans le stator est $P_{js}=3R\ I^2$

NB : Si Ra est la résistance mesurée entre deux entrées de la plaque à bornes alors quel que soit le couplage choisi et pour un courant par phase I les pertes joules statoriques sont : $Pjs=3/2\ Ra\ I^2$.

- dans les paliers à cause du frottement entre des pièces mécaniques.
- dans la masse métallique à causes des effets de l'hystérésis et des courants de Foucault qui sont des pertes qui dépendent de la fréquence du réseau.

Pour une fréquence donnée, les deux dernières pertes sont pratiquement constantes et ne dépendent pas de la charge. Elles ne sont pas calculables. Elles sont déterminées par des essais.

Activité
Réaliser l'activité N°4 du TP-B3 dans le manuel d'activités
Il s'agit de déterminer pour un moteur asynchrone tournant en charge :
- la vitesse de rotation et le glissement
- la puissance active absorbée par le moteur
- les pertes joules statoriques
- la puissance transmise
- le couple électromagnétique
- la puissance utile.

F- FONCTIONNEMENT EN CHARGE

1- Conditions de fonctionnement :

En charge, le moteur entraîne en rotation un mécanisme quelconque
(**Exemple** : compresseur de la cintreuse).
Le glissement est important et le moteur tourne à une vitesse inférieure à la vitesse de synchronisme :

$g > 0$ et donc $n < n_S$

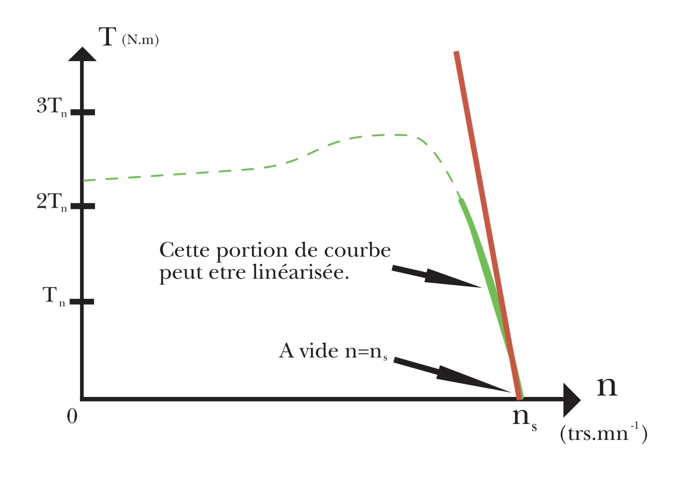
Broyeur de déchets plastiques

La puissance utile est fonction de la charge entraînée. Si la charge présente un couple résistant T_r, la puissance mécanique utile **Pu** est alors :

$$Pu = T_r.\Omega$$

(avec T_r en Newton et Ω en rd.s^{-1})

Lorsqu'on modifie la charge, le couple utile et la vitesse de rotation du rotor varient simultanément. La fonction qui lie ces deux grandeurs physiques est donnée par la courbe caractéristique suivante appelée **caractéristique mécanique.**



Légende:

- Tn Couple nominal
- – – – Couple du moteur
- Zone de fonctionnement du moteur
- Zone de fonctionnement linéarisée

NB: On constate que le couple de démarrage du moteur est fort: le moteur asychrone peut alors démarrer en charge.

Le facteur de puissance en charge est élevé. Le stator appelle un courant actif important et le moteur absorbe davantage de puissance active :

$Pa = \sqrt{3}.U.I.\,Cos\varphi$ avec U tension entre deux bornes du moteur en V
I courant de ligne en A

Les pertes joules dans le stator augmentent car le courant dans les bobinages est important.
Les pertes constantes (mécaniques **Pm** et magnétiques P_{fs} au stator et P_{fr} au rotor) restent les mêmes qu'à vide.
Le stator transmet au rotor une puissance électromagnétique appelée **puissance transmise qui** est notée P_{tr} : $P_{tr} = P_a - p_{js} - p_{fs}$.

Le stator exerce sur le rotor un couple électromagnétique appliquant ainsi un ensemble de forces sur ses conducteurs qui le font tourner à une vitesse légèrement inférieure à la vitesse de synchronisme. Ce couple doit être suffisant pour entraîner la charge accouplée au moteur.

$$Tem = \frac{Ptr}{\Omega s}$$

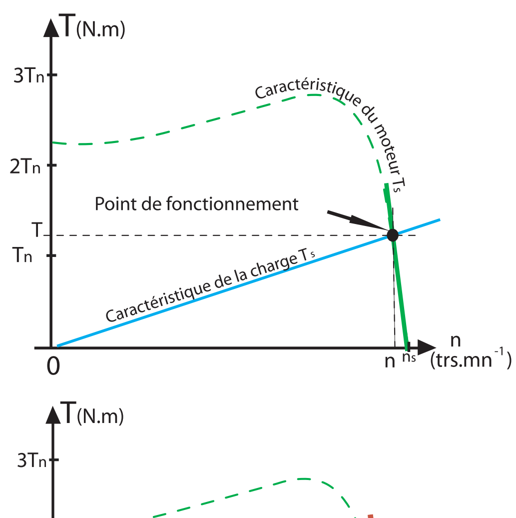
avec Tem : couple électromagnétique en Nm
Ptr : puissance transmise en W
Ωs : vitesse angulaire de synchronisme en $rd.s^{-1}$

La puissance électromagnétique **Pem = Tem. Ω** est transformée en puissance mécanique dont une faible partie est perdue par frottement au niveau des paliers et toute la puissance restante devient disponible sur l'arbre du moteur.

Elle est appelée puissance mécanique utile. $P_u = P_{em} - p_m$

2- Point de fonctionnement :

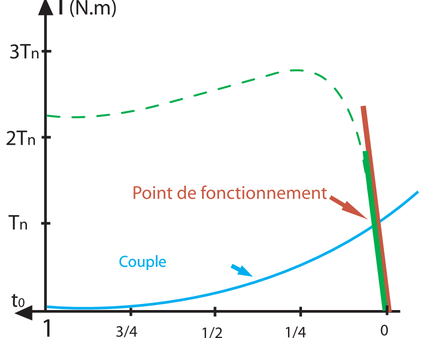
Pour que le moteur puisse fonctionner normalement le point de fonctionnement stable doit se situer sur la partie linéaire de la caractéristique mécanique. Le point de fonctionnement correspond à l'intersection de cette partie linéaire **Tu = f(n)** avec la caractéristique de la charge **Tr = f(n)**.
Le point de fonctionnement peut être déterminé graphiquement ou par calcul en mettant en équation les parties linéaires des deux caractéristiques.



Exemple : Cas d'une charge ayant un couple résistant proportionnel au carré de la vitesse. $Tr = c.n^2$

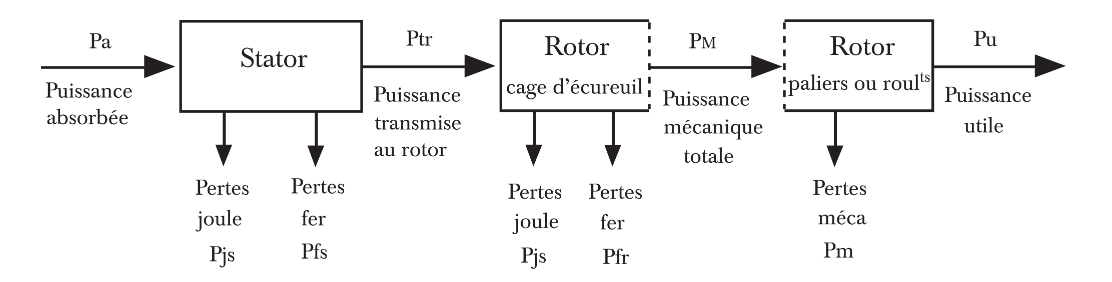
$$Tu = Tr \Rightarrow a.n + b = c.n^2$$

Finalement, il faut résoudre une équation du second degré : $c.n^2 - a.n - b = 0$
En étudiant les deux solutions trouvées, on retiendra celle du cadran positif.



G- BILAN ENERGETIQUE D'UN MOTEUR ASYNCHRONE

En remplaçant le stator et le rotor par deux modèles fonctionnels équivalents, un moteur asynchrône peut être dans ce cas représenté par le schéma suivant :





Le rendement du moteur est donné par le rapport de la puissance utile Pu disponible sur l'arbre à la puissance Pa :

Puissance absorbée	$Pa = \sqrt{3}.U.I.\cos\varphi$
Puissance transmise	$Ptr = Pa - p_{js} - p_{fs} = T_{em}.\Omega_s$
Pertes joules stator	$p_{js} = \frac{3}{2}.R_a.I^2 \quad \forall$ le couplage
Pertes fer stator	P_{fs} est déterminée par un essai à vide
Puissance mécanique	$P_M = P_{tr} - p_{jr} - p_{fr} = T_{em}.\Omega$
Pertes fer rotor	p_{fr} négligée (fréquence des courants rotoriques très faible)
Pertes joules rotor	$p_{jr} + p_{fs} = P_{tr} - P_M \xrightarrow{p_{fs}=0} p_{jr} = P_{tr} - P_M$
	$p_{js} = T_{em}.\Omega_s - T_{em}.\Omega = T_{em}.(\Omega_s - \Omega) = \frac{T_{em}}{\Omega_s}.\frac{(\Omega_s - \Omega)}{\Omega_s} = P_{tr}.g$
Puissance utile	$P_u = P_M - p_m = Tu.\Omega$
Pertes mécaniques	p_M est déterminée par un essai à vide η

L'établissement du bilan permet de calculer : $\eta = \frac{Pu}{Pa} = 1 - \frac{\sum pertes}{Pa}$

G- VARIATION DE VITESSE D'UN MOTEUR ASYNCHRONE

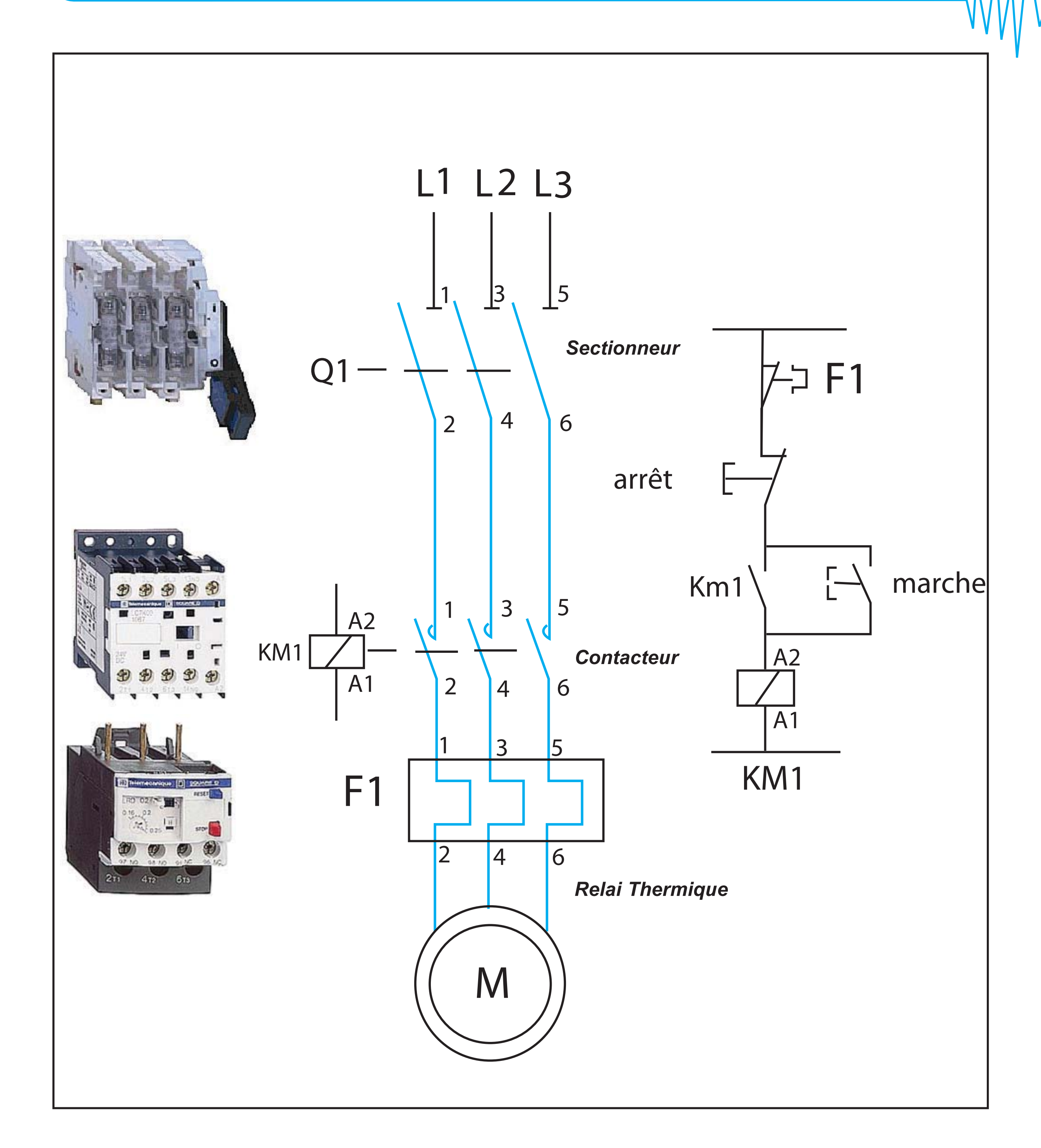
1- Domaine d'utilisation des moteurs asynchrones :

Le moteur asynchrone triphasé, dont la puissance varie de quelques centaines de watts (perceuse d'établi) à plusieurs mégawatts (Traction ferroviaire) est le plus utilisé de tous les moteurs électriques. Son rapport coût/puissance est le plus faible.
Associés à des onduleurs de tension, les moteurs asynchrones de forte puissance peuvent fonctionner à vitesse variable dans un large domaine (les derniers TGV de l'Europe...).
Toutefois l'emploi de ce type de moteur est à éviter en très forte puissance (P>10MW) car la consommation de puissance réactive est alors un handicap.

2- Fonctionnement à vitesse constante :

Le moteur est relié au réseau par un certain nombre de dispositifs de sécurité et de commande :
- Le sectionneur d'isolement avec fusibles permet de déconnecter le moteur du réseau pour des opérations de maintenance par exemple. Il protège également le dispositif en aval contre les risques de court circuit grâce aux fusibles.
- Le contacteur permet d'alimenter le moteur avec une commande manuelle ou automatique avec un automate programmable.
- Le relais thermique protège le moteur contre les surcharges de courant, l'intensité maximale admissible est réglable.

Remarques : en électroménager (exemple : lave-linge) la vitesse des moteurs asynchrones n'est pas réglée par un onduleur, mais ces moteurs possèdent plusieurs bobinages. Il est alors possible de changer le nombre de paires de pôles et donc la vitesse.



3- Fonctionnement à vitesse variable :

Bien que sa vitesse soit liée à la fréquence du réseau, le moteur asynchrone reste toujours d'actualité car l'électronique permet maintenant de faire varier sa fréquence de rotation. Pour faire varier celle-ci, il faut modifier la fréquence de rotation du champ magnétique et donc la fréquence du courant d'alimentation.

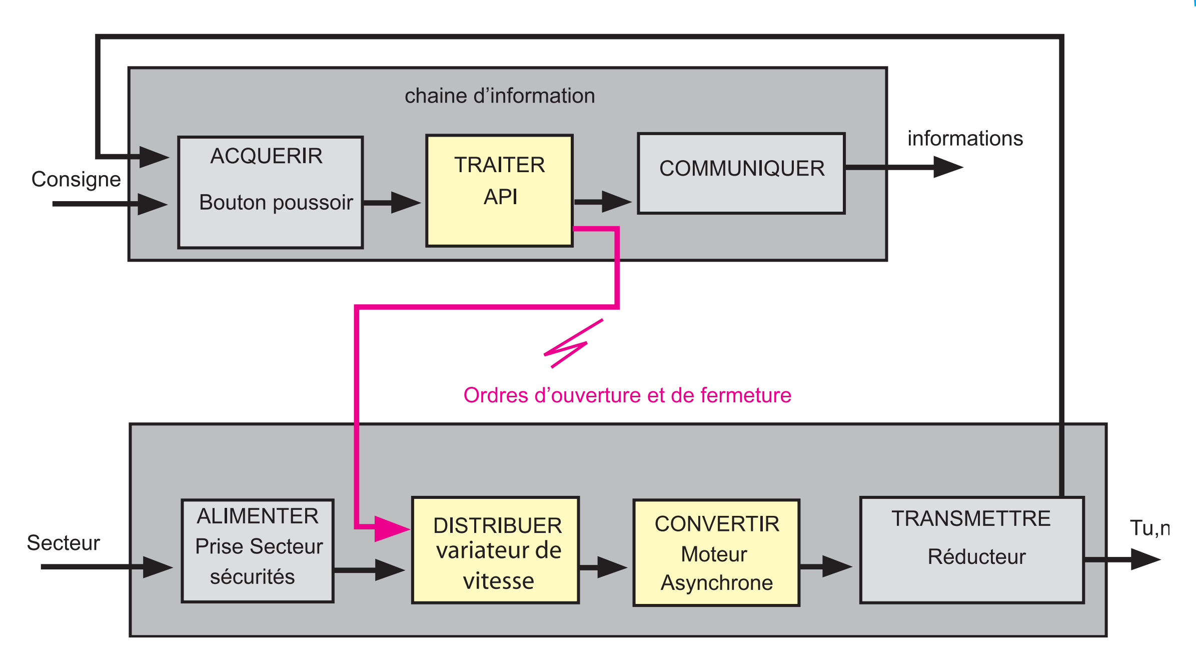


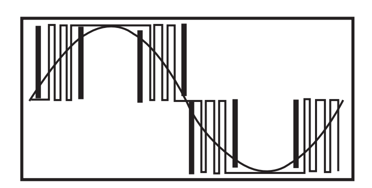
Schéma fonctionnel d'un système qui utilise un variateur de vitesse pour moteur asynchrone

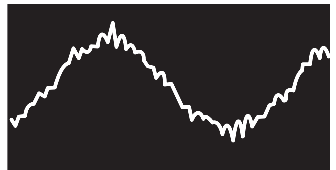


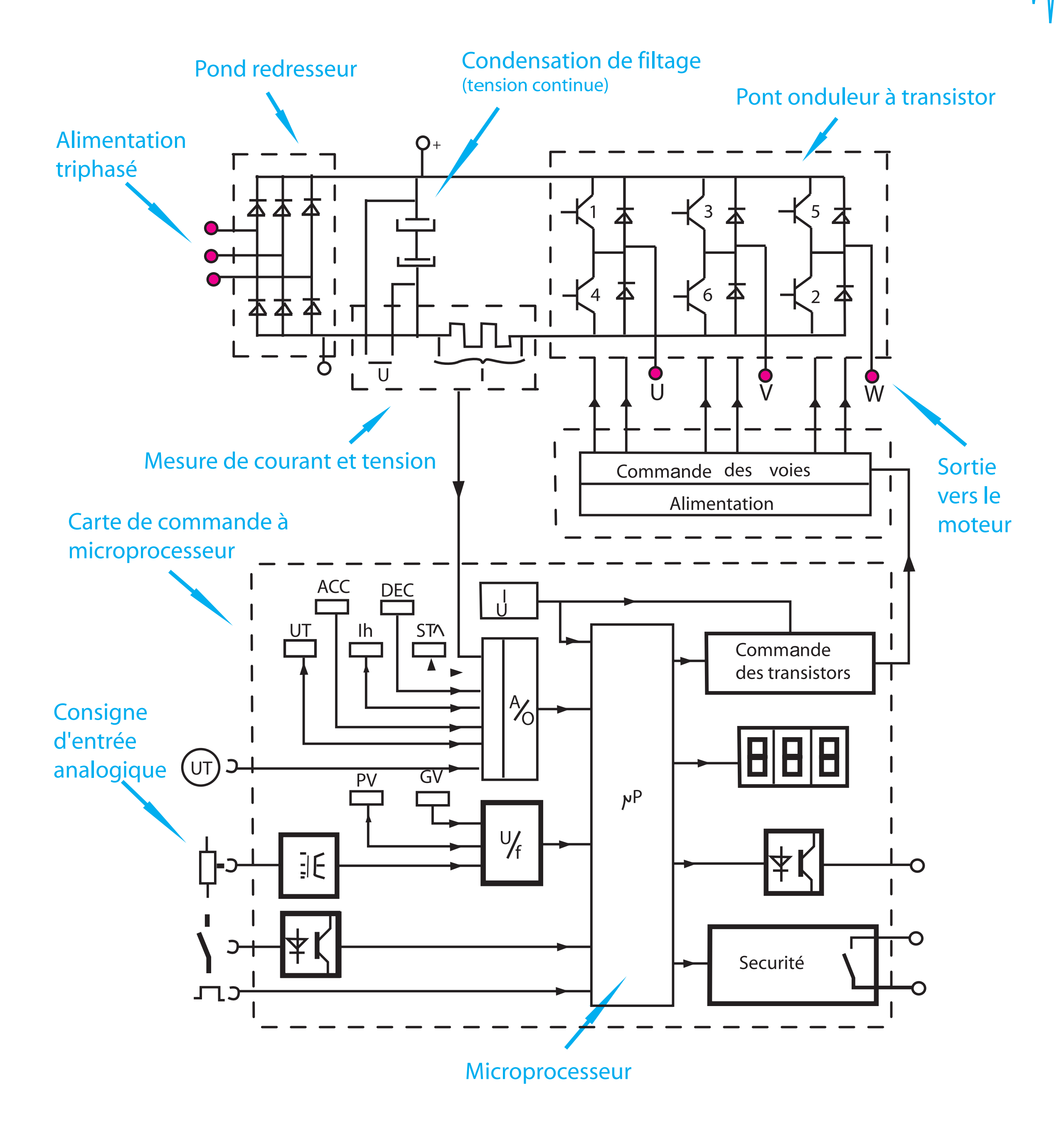
Les variateurs de vitesse sont des variateurs de fréquence.
Ils permettent
-Une gamme de vitesses de 5% à 200% de la vitesse nominale.
-Une conservation du couple sur toute la gamme de vitesses.
-Des rampes d'accélération et de décélération
-Deux sens de rotation.

La consigne de vitesse est en général fournie sous forme d'une tension de 0 à 10V par exemple Une protection du moteur est intégrée au variateur.
Le courant électrique issu du réseau est dans un premier temps converti en courant continu, il est ensuite reconverti en courant alternatif par un onduleur mais avec une fréquence réglable. Il est ainsi possible de convertir du monophasé en triphasé si c'est nécessaire.

L'onduleur travaille en hacheur, il va moduler le courant par largeur d'impulsions (PWM), le courant résultant est proche d'une sinusoïde.







H- EXERCICE D'APPLICATION :

1- Exercice résolu :

Un moteur asynchrone triphasé, dont le stator est monté en étoile, est alimenté par un réseau 380 V entre phase **50 Hz**. Chaque enroulement du stator a une résistance **R = 0,4 W.**

Lors d'un essai à vide, le moteur tournant pratiquement à **1500 tr/min**, la puissance absorbée est de P_V **= 1150 W**, le courant par fil de ligne est I_V **= 11,2 A**.

Un essai avec la charge nominale sous la même tension de **380 V**, **50 Hz**, a donné les résultats suivants :

- glissement : **4%**,
- puissance absorbée : **18,1 kW**,
- courant en ligne : **32 A**.

1) Essai à vide :

a/ Calculer les pertes par effet Joule dans le stator lors de l'essai à vide. Que peut-on dire des pertes par effet Joule dans le rotor lors de cet essai?

b/ En déduire les pertes dans le fer sachant que les pertes mécaniques valent **510 W**.

2) Essai en charge :

a/ Calculer le facteur de puissance nominal et la fréquence nominale de rotation.

b/ Calculer la fréquence des courants rotoriques pour un glissement de 4%. Que peut-on en déduire pour les pertes dans le fer du rotor?

3) Calculer les pertes par effet Joule dans le stator et dans le rotor en charge nominale.

4) Calculer la puissance utile et le rendement du moteur en charge nominale.

5) Calculer le moment du couple utile nominal.

Corrigé :

1) Essai à vide :

a/ Pertes Joule

A vide, la puissance absorbée se décompose en:

Pv = Pfs + Pjs + Pm

Pfs: pertes fer au stator

Pjs0: pertes Joule au stator à vide

pm: perte mécaniques

Les pertes Joule au rotor sont proportionnelles au glissement et à la puissance transmise Ptr

Pjs = gPtr = g (Pabs - Pfs - Pjs)

A vide le glissement est très faible, la vitesse de rotation du rotor est quasiment égale à la vitesse de synchronisme, et la puissance transmise est faible (Puissance utile nulle). A vide, les pertes Joule au rotor sont donc négligeables.

b/ Pertes fer et pertes mécaniques :

A vide les pertes Joule au stator s'exprime par:

Pjs0 = (3/2)RaI02 = 150,5 W

où Ra = 2R = 0,8 W est la résistance mesurée entre phase au stator.

On a donc : Pfs + Pm = Pv - Pjs0 = 999,5 W

Pm = 510 W, d'où : Pfs = 489,5 W

Pour ce qui est des pertes fer au rotor, que ce soit en charge ou à vide, elles sont fonction de la tension au rotor et de la fréquence des courants rotoriques. La fréquence des courants au rotor étant très faible (f_{rotor} = g f_{stator}) et celui ci étant en court-circuit les pertes fer au rotor peuvent être négligées.

2) Essai en charge :

a/ Facteur de puissance et vitesse de rotation

A partir de la définition de la puissance active en triphasé on déduit

Le glissement étant défini par g = (N - N') / N

on a N' = (1 - g)N = 1440 tr/mn

b/ Fréquence des courants rotoriques

$$f_{rotor} = g\ f_{stator} = 2\ Hz$$

Concernant les pertes fer, la remarque de la question précédente reste valable, elles sont toujours négligeables.

3) Pertes Joule au stator et au rotor :

Pertes Joule au stator

Elles sont données par : Pjs = (3/2)RaI2 = 1228,8 W

Pertes Joule au rotor

Elles sont proportionnelles à la puissance transmise

Pjr = gPtr = g (Pabs - Pfs - Pjs) = 655,3 W

4) Puissance utile et rendement en charge :

La puissance utile est donnée par

Pu = Pabs - Pfs - Pjs - Pjr - Pm
= (1 - g) (Pabs - Pfs - Pjs) - Pm
= (1 - g) Ptr -Pm
= 15216,4 W

Le rendement est donc η = Pu / P = 0,84

5) Moment du couple utile

Par définition il est donné par Cu = Pu / 2πN' = 100,9 Nm (N' en tours /seconde)

2- Exercice à résoudre :

Exercice 1 :

Un moteur asynchrone triphasé a les caractéristiques suivantes :
- tension d'alimentation : 115/200 V. Rotor à cage moteur tétrapôlaire.
- fréquence : 400 Hz.
- vitesse nominale : 11 500 tr/min.
- puissance absorbée en charge nominale: 4 200 W, cosφ = 0,6.
- résistance de chaque enroulement du stator: Rs = 0,16 Ω.

Le moteur est alimenté par un réseau triphasé 200 V, 400 Hz. Il entraîne sa charge nominale.

1) Quel est le couplage à adopter ?
2) Quel est le glissement ?
3) Quelle est l'intensité du courant absorbé en ligne ?
4) Quelles sont les pertes joule au stator ?

5) Déterminer le rendement sachant que les pertes fer au stator sont de 350 W et que l'on néglige les pertes fer au rotor ainsi que les pertes mécaniques ?
6) Quel est le couple utile ?

Exercice 2 :

Un moteur asynchrone triphasé à rotor à cage d'écureuil est alimenté par un réseau triphasé 50 Hz, 220/380 V. Pour le stator, le couplage des enroulements est fait en étoile. Chaque enroulement du stator a une résistance Rs = 0,285Ω.

On réalise un essai à vide : le moteur tourne pratiquement à la vitesse de synchronisme (N=3000 tr/min). La puissance absorbée à vide est P_0 = 3 kW et le courant de ligne est I_0 = 25 A.

1) Calculer le nombre de pôles du stator et le facteur de puissance à vide.
2) On supposera les pertes mécaniques constantes sont égales à 1233 W dans la suite du problème. Que peut-on dire des pertes joules au rotor (Pjr) ?
3) Calculer les pertes joules stator (Pjs) et les pertes fer stator (Pfs) lors de cet essai à vide.
On réalise un essai en charge, les résultats sont les suivants :
- glissement : 7%,
- puissance absorbée : 24645 W,
- courant en ligne : 45 A.

4) Calculer le facteur de puissance, la vitesse de rotation du rotor, la fréquence des courants rotoriques lors de cet essai.
5) Faire un bilan de puissance. Calculer Pjs et la puissance transmise au rotor Ptr. En déduire Pjr lors de cet essai en charge.
6) Calculer la puissance utile Pu, le rendement du moteur, le couple utile Tu et le couple électromagnétique Te.

Faire un bilan de puissance et indiquer le nom et la valeur de chaque puissance.

Le moteur entraîne une machine dont la caractéristique mécanique est une droite d'équation :

$$Tr = 2/100\ N' + 40 \quad (N' \text{ en tr/min})$$

7) Calculer la vitesse du groupe (moteur + machine d'entraînement) sachant que la caractéristique mécanique du moteur est une droite en fonctionnement normal (donc valable pour l'essai en charge effectué précédemment).

LES MOTEURS À COURANT CONTINU

A- MISE EN SITUATION :

1- Présentation du système : Trottinette électrique à 2 roues

La pollution atmosphérique issue de nos chères voitures favorise le développement de l'effet de serre : les particules de **gaz carbonique** (CO_2) qui s'échappent des pots d'échappement retiennent les rayonnements solaires à l'intérieur de l'atmosphère terrestre, participant grandement au **réchauffement de la planète**.
Et pourtant, saviez-vous que **50 %** des trajets effectués en voiture particulière font moins de 3 kilomètres.
Le véhicule électrique contribue très efficacement à la **diminution de la pollution** urbaine car il n'émet aucun polluant à l'usage.



Principales caractéristiques :
* Cadre en tube en aluminium, soudé.
* Châssis et console en matière synthétique.
* Unité de direction rabattable.
* Largeur des pneus 9 cm. Chambre à air avec valve comme sur les voitures.
* Hauteur du guidon 100 à 115 cm.
* Poids spécifique 22 kilos.
* Charge utile 100 kilos.

Performances
Vitesse 20 km/h.
Autonomie par charge de batterie jusqu'à 45 minutes en vitesse maximale.



Equipement
Batterie, chargeur de batterie et câble de fermeture inclus.

2- Problèmes posés :

a) Quel est l'actionneur capable de fournir de l'énergie mécanique à partir de l'énergie électrique fournie par une batterie d'accumulateur ?
b) Comment choisir et dimensionner cet actionneur ?
c) Comment protéger et commander cet actionneur ?

B- RAPPEL :

1- Fonctionnement à vide :

Activité

Réaliser l'activité N°1 du TP-B4 dans le manuel d'activités

Il s'agit de faire tourner à vide un moteur à courant continu et de mesurer :
- la vitesse de rotation;
- l'intensité du courant absorbé par l'induit
- l'intensité du courant absorbé par l'inducteur
- la tension appliquée aux bornes de l'induit
- la tension appliquée aux bornes de l'inducteur

Le moteur est désaccouplé de toute charge. Il fonctionne sous tension nominale.

La puissance utile est nulle

$$Pu = 0 ;$$

La puissance perdue par effet joule :
- dans le circuit inducteur (R étant la résistance du bobinage inducteur mesurée à chaud)

$$P_{jex} = r.i_{ex}^2 = u.i_{ex}$$

- dans le circuit induit (R étant la résistance du bobinage d'induit mesurée à chaud)

$$P_{ja0} = R.I_{a0}^2$$

La puissance absorbée par l'induit à vide :

$$P_{a0} = U.I_{a0}$$

Les pertes constantes (par frottement au niveau des paliers et par hystérésis)

$$P_{cte} = P_{mec} + P_f$$
$$P_{cte} = U.I_{a0} - P_{ja0}$$
$$P_{cte} = U.I_{a0} - R.I_{a0}^2$$
$$P_{cte} = I_{a0} (U - R.I_{a0})$$
$$P_{cte} = I_{a0} E'_0 \qquad (E'_0 : \text{f.c.e.m. à vide})$$

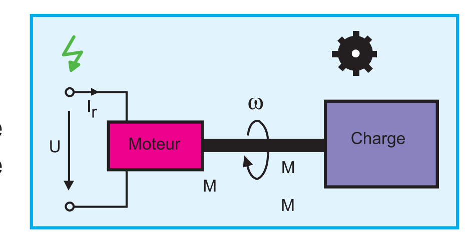
Activité

Réaliser l'activité N°2 du TP-B4 dans le manuel d'activités

Il s'agit de faire tourner un moteur à courant continu et de mesurer pour une charge donnée (proche de la nominale) :
- la vitesse de rotation;
- l'intensité du courant absorbé par l'induit
- la tension appliquée aux bornes de l'induit

2- Fonctionnement en charge :

Le moteur est maintenant chargé, c'est-à-dire que l'arbre de ce dernier entraîne une charge résistante qui s'oppose au mouvement du rotor.



Le moteur restitue une puissance mécanique importante, combinaison du couple utile et de la fréquence de rotation.

$$Pu = Tu.\Omega$$

La puissance perdue par effet joule :

- dans le circuit inducteur :

elle est la même qu'à vide

- dans le circuit induit (I_a étant le courant absorbé en charge) :

$$P_{ja} = R.I_a^2$$

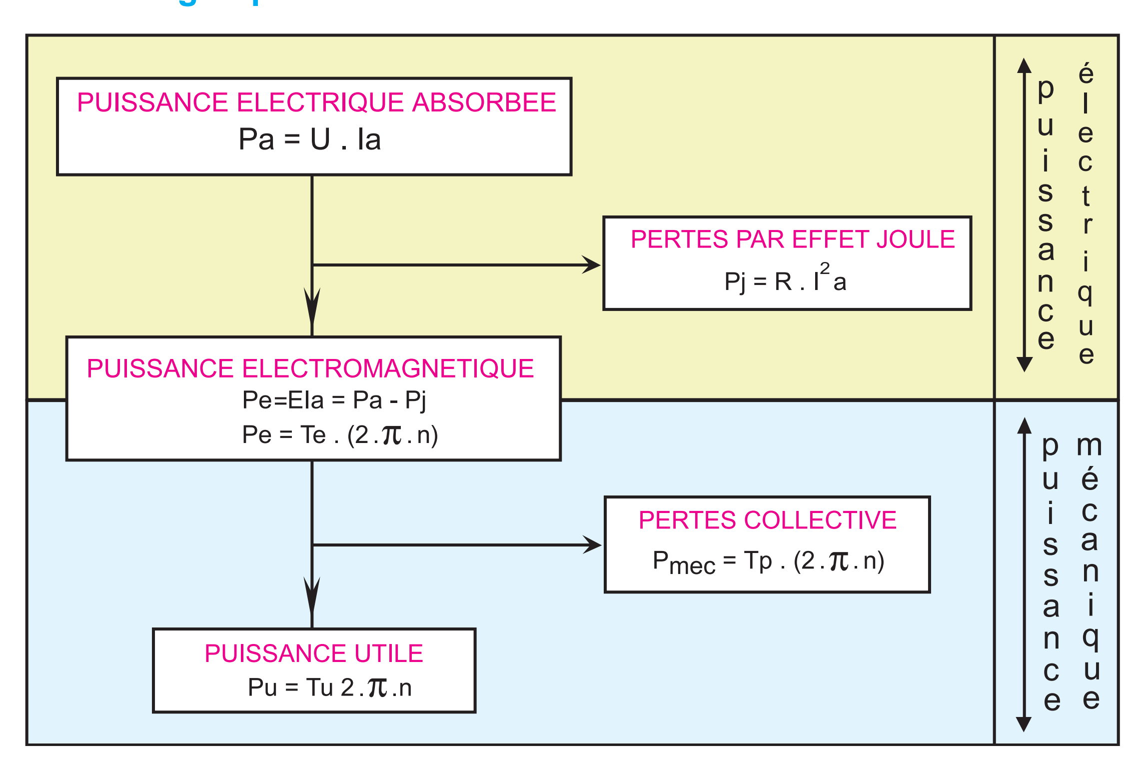
La puissance absorbée par l'induit en charge :

$$P_a = U.I_a$$

Les pertes constantes (par frottement au niveau des paliers et par hystérésis)

elles sont les mêmes qu'à vide

3- Bilan énergétique :



Le rendement se définit comme le rapport de la puissance utile fournie, à la puissance absorbée. Il est de l'ordre de 60 à 80% pour les machines de moyennes puissances et 90% pour les plus puissantes.

$$\eta = \frac{Pu}{Pa}$$

C- CARACTÉRISTIQUES D'UN MOTEUR A COURANT CONTINU :

Pour prévoir le fonctionnement en charge d'un moteur à courant continu, il faut nécessairement connaître son comportement pour des charges différentes. Il est donc nécessaire de tracer une description comportementale de la machine appelée couramment **caractéristique**.

1- Caractéristique de couple :

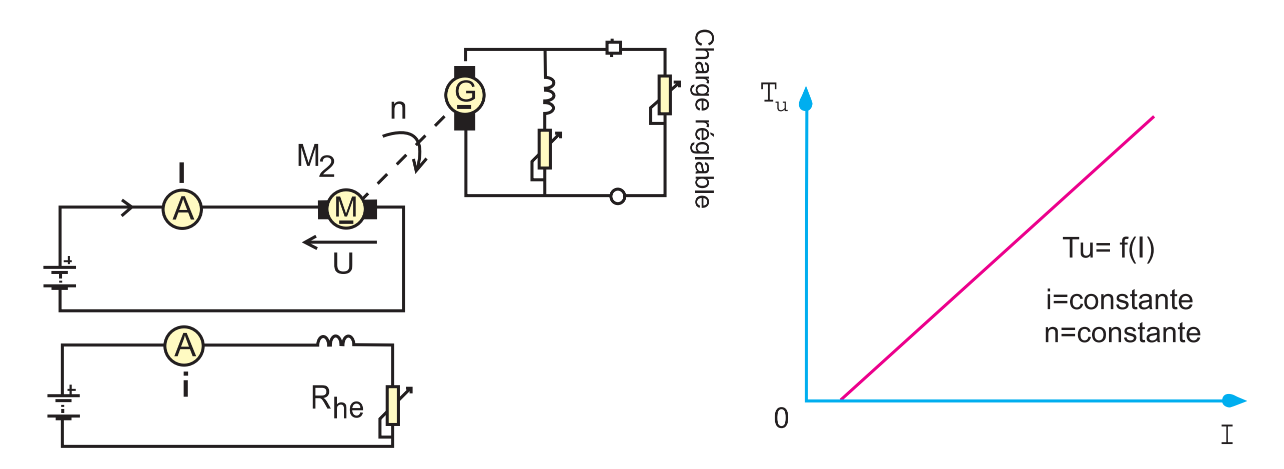
Activité

Réaliser l'activité N°3 du TP-B4 dans le manuel d'activités

Il s'agit de faire tourner un moteur à courant continu, de varier sa charge et de mesurer pour différentes situations :

- la vitesse de rotation
- l'intensité du courant absorbé par l'induit
- le couple fourni par le moteur (par acquisition numérique ou par mesure avec dynamo-balance).

C'est la courbe représentative du couple en fonction du courant absorbé par le moteur pour des charges progressives.



Chaque conducteur est le siège d'un effort créant un couple sur l'arbre. Leur somme génère le couple électromagnétique Tem.

Suivant la relation issue de la loi de Laplace, le couple dépend du flux sous un pôle (ϕ en Wb), du courant total d'induit (I en A) et du nombre de conducteurs (N).

D'où :

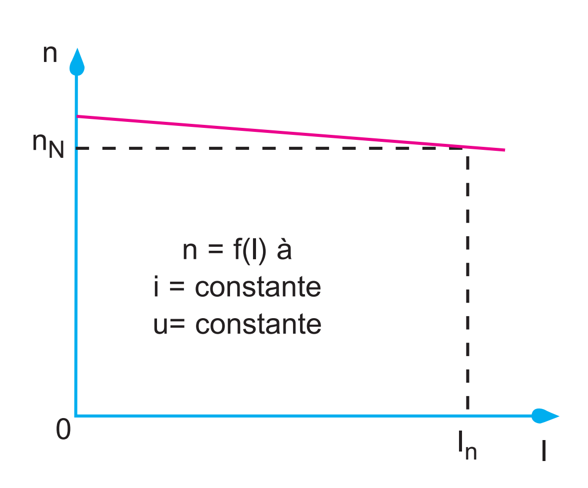
$$Tem = \frac{Pem}{\Omega} = \frac{E'I}{\Omega} = \frac{N_n \Phi I}{2\pi n} = \frac{N\Phi I}{2n} = KI.$$

$Tu = Tem - T_1$ avec T_f est le couple de frottement.

2- Caractéristique de vitesse :

C'est la courbe qui représente la vitesse en fonction du courant absorbé par le moteur pour des charges progressives.

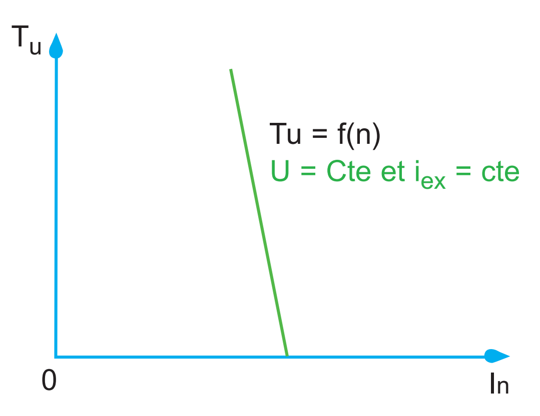
$$E' = Kn$$
$$E' = U - RaI$$
$$\text{donc : } Kn = U - RaI$$
$$\text{d'où : } n = \frac{U - RaI}{K}$$
$$n = \frac{U}{K} - \frac{Ra}{K}.I = A - BI$$



3- Caractéristique mécanique :

Le couple et la vitesse dépendent tous les deux du courant absorbé par l'induit, si l'excitation et la tension U sont maintenues constantes :

$$Tem = KI$$
$$n = \frac{U}{K} - \frac{Ra}{K}.I = A - BI \Rightarrow I = A - \frac{n}{B}$$
$$Tem = K\left(A - \frac{n}{B}\right) = AK - \frac{K}{B}n = a - bn$$

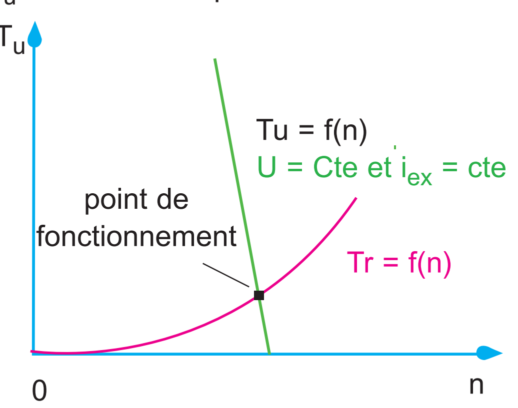


4- Point de fonctionnement :

Une charge oppose au moteur un couple résistant T_r. Pour que le moteur puisse entraîner cette charge, le moteur doit fournir un couple utile T_u de telle sorte que :

$$T_u = T_r$$

Cette équation correspond à un point commun entre la caractéristique électromécanique et la courbe du couple résistant. Le fonctionnement du moteur ne peut être stable que lorsque les grandeurs installées correspondent bien à ce point.



D- VARIATION DE VITESSE D'UN MOTEUR A COURANT CONTINU

Activité
Réaliser l'activité N°3 du TP-B4 dans le manuel d'activités
Il s'agit de faire tourner un moteur à courant continu, de varier sa vitesse et de mesurer pour différentes situations :
- la vitesse de rotation;
- l'intensité du courant absorbé par l'induit
- le couple fourni par le moteur (par acquisition numérique ou par mesure avec dynamo-balance)

1- Pourquoi varier la vitesse ?

L'augmentation du coût de l'énergie nous amène à faire des recherches en vue d'améliorer le rendement énergétique des différents procédés industriels. Les entraînements de puissance peuvent être améliorés par l'utilisation de machines électriques à vitesse variable.

La variation de vitesse d'un moteur à courant continu peut être obtenue par action sur la tension de l'induit (U) ou sur le flux inducteur (Φ).

$$n = \frac{U - R.I}{N\,\Phi}$$

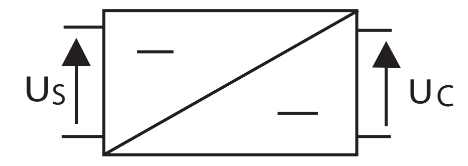
La variation de la vitesse par action sur la tension aux bornes de l'induit U permet d'obtenir une vitesse variable de zéro jusqu'à la valeur nominale de la vitesse, tout en ayant un couple maximal important en tout point de fonctionnement. Ce procédé est le plus utilisé et nécessite une source de tension continue réglable.



2- Principe de fonctionnement d'un variateur de vitesse : hacheur série

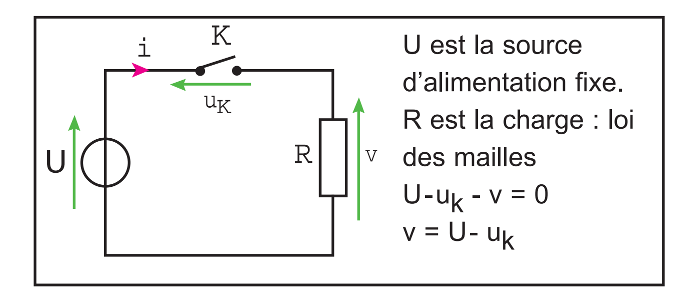
Les hacheurs permettent le transfert d'énergie entre une source de **tension continue constante Us** et une charge fonctionnant sous **tension continue** éventuellement **réglable Uc**.
Le hacheur est un convertisseur continu continu. Il permet d'obtenir du continu variable à partir d'une source continue fixe (pile, alimentation stabilisée, réseau redressé…).

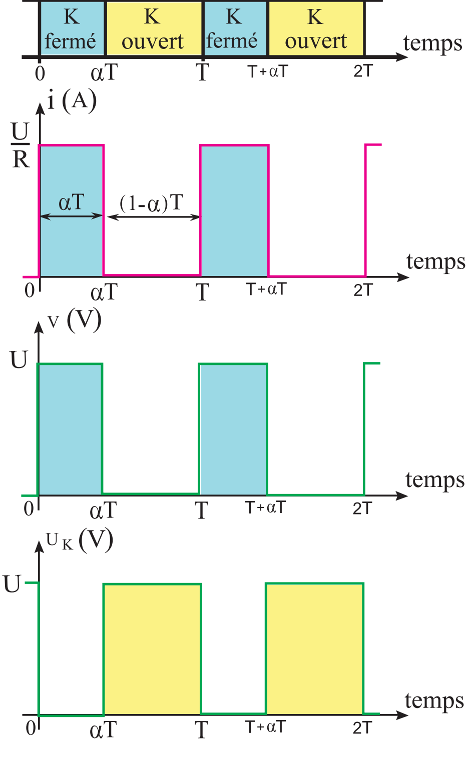
Uc < Us Hacheur abaisseur de tension.
Uc > Us Hacheur élévateur de tension.



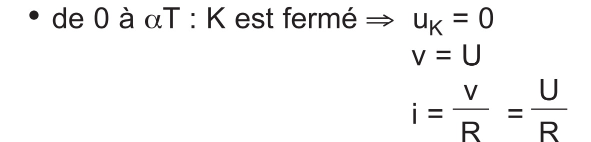
3- Montage de principe : débit sur une charge résistive

(en réalité l'interrupteur est remplacé par un transistor)





On choisit une période T et une fraction **α** de cette période. **α** s'appelle le **rapport cyclique**, $0 < \alpha < 1$, sans dimension.



- de 0 à αT : K est fermé $\Rightarrow$ $u_K = 0$
  $v = U$
  $i = \frac{v}{R} = \frac{U}{R}$

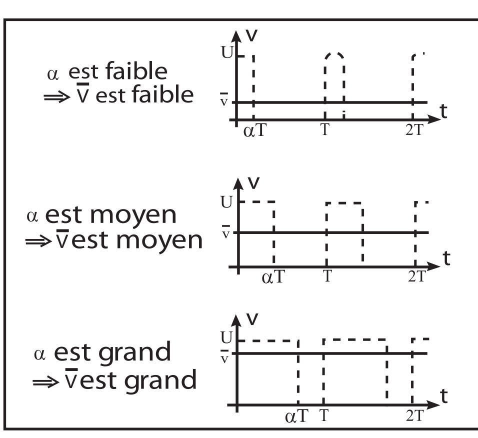
- de αT à T : K est ouvert $\Rightarrow$ $i = 0$
  $v = Ri = 0$

4- Interprétation :

✔ La tension de sortie du hacheur (tension v) n'est pas continue mais toujours positive. Lorsque la période est assez faible (fréquence de 100 à 1000 Hz) la charge ne « voit » pas les créneaux mais la valeur moyenne de la tension.

✔ le rapport cyclique a peut être réglé. Par conséquent la valeur moyenne v (ou < v >) de v va varier.

✔ il s'agit d'un hacheur série car l'interrupteur K est monté en série entre la source et la charge.



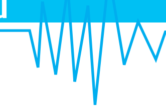
5- Valeur moyenne de la tension en sortie du hacheur :

Exprimons la valeur moyenne de u en fonction du rapport cyclique α.
Pour cela nous calculons sa valeur moyenne sur une période :

$$\bar{v} = \frac{\alpha T.U + (1-\alpha)\,T.0}{T}$$

$$\bar{v} = \alpha U$$

Valeur moyenne : $\bar{v} = \alpha.U$



6- Débit sur une charge inductive :

a/ Propriété des inductances

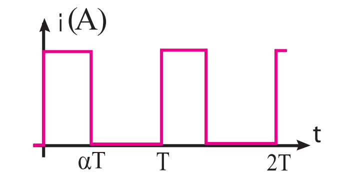
Equation fondamentale : $u_L = L.\dfrac{di}{dt}$

De cette équation nous pouvons démontrer les propriétés ci-dessous :

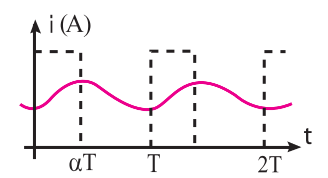
- En régime continu établi : l'inductance se comporte comme un court-circuit.
- En régime périodique établi : la tension moyenne est nulle : $u_L = L.\dfrac{di}{dt}$
- En régime quelconque, d'une façon générale :
  - le courant dans une inductance ne peut pas subir de discontinuité.
  - l'inductance s'oppose aux variations du courant qui la traverse, et ce d'autant plus que :
    - L est grand ;
    - la tension aux bornes de l'inductance est plus faible.

Par conséquent : Une inductance lisse le courant.

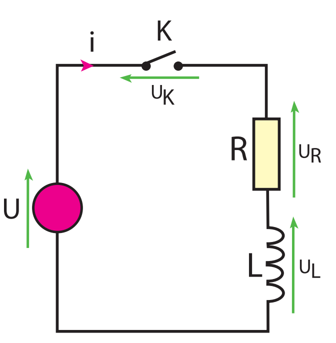
courant pour une charge résistive



courant pour une charge inductive :

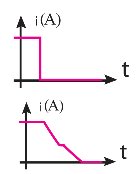


7- Problème lié aux charges inductives :



A la fermeture de K le courant s'établit.
A l'ouverture de K deux phénomènes contradictoires ont lieu :

- la commande qui veut annuler subitement le courant
- la bobine qui ne peut subir de discontinuité de courant

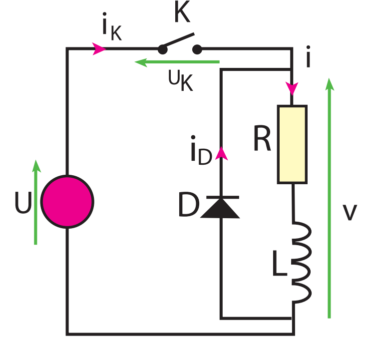


C'est la bobine qui « gagne » en provoquant un arc électrique aux bornes de l'interrupteur pour maintenir le courant.

Conséquence :

L'interrupteur qui est en réalité un transistor (ou autre composants commandé tel que thyristor ...) subit alors à chaque blocage une surtension qui peut être destructrice. Il faut prévoir un système qui permette le blocage normal du transistor.

8- Solution et analyse du fonctionnement :



Rappel :

Loi des mailles

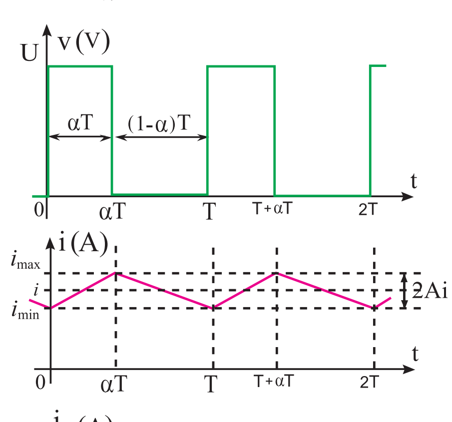
$U - u_K - v = 0$

$v = U - u_K$

Loi des noeuds

$i = i_K - i_D$

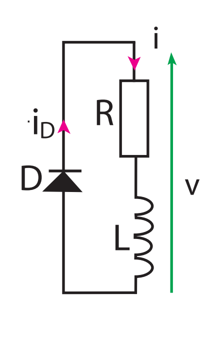
Analyse du fonctionnement

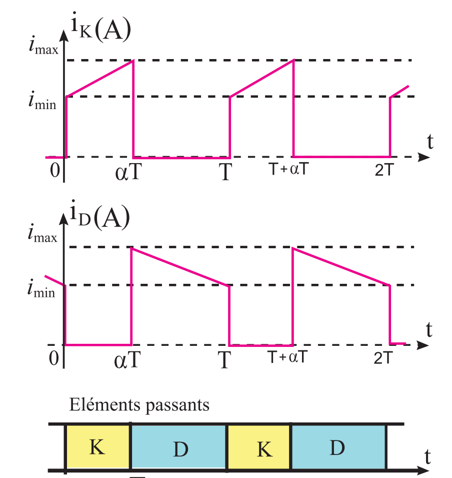


• **de 0 à αT : K est fermé.**

La source U alimente la charge.
Le courant ne peut pas passer par la diode.
$u_K = 0 \Longrightarrow v = U$
$i = i_K$ et $i_D = 0$
Le courant augmente progressivement (la pente dépend de la valeur de L).

• **de αT à T : K est ouvert.**





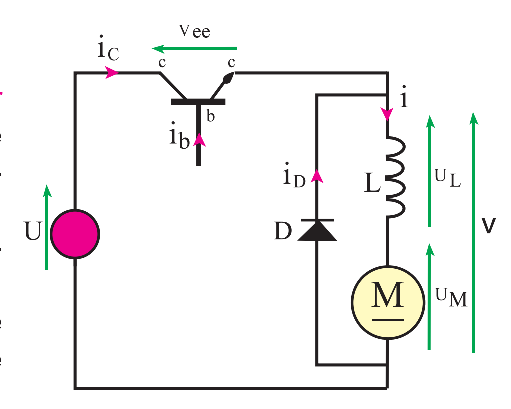
La bobine maintient le courant à travers la diode.
$v \approx 0 \Longrightarrow u_K = U$
$i = i_D$ et $i_K = 0$
Comme la charge n'est pas alimentée, le courant diminue progressivement.

Conclusions :

- A l'ouverture de K, il n'y aura pas d'étincelle puisque le courant imposé par la bobine pourra passer par la diode.
- D est appelée diode de roue libre car elle est active lorsque la charge n'est pas alimentée. Elle est nécessaire pour un bon fonctionnement du montage.
- La bobine lisse le courant. Plus l'inductance est grande, plus Δi sera petit (voir les oscillogrammes).

9- Application au moteur :

Le hacheur série est souvent employé pour commander un moteur à courant continu. On rappelle que la vitesse d'un tel moteur est proportionnelle à la tension d'alimentation.
Pour un bon fonctionnement du moteur, il est préférable que le courant soit le plus régulier possible, d'où la présence d'une bobine de lissage. Si son inductance est suffisamment grande, on pourra considérer le courant comme constant ($\Delta i \approx 0$).



$$\overline{v} = R_M . i + u_b + E$$

Puisque la valeur moyenne de la tension aux bornes d'une bobine est nulle et puisque Rb est négligeable alors on peut écrire : $u_b = 0$

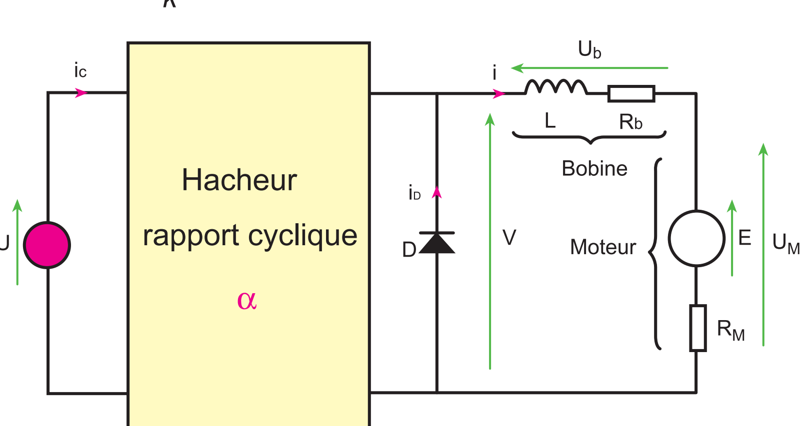
Donc : $\overline{v} = R_M . i + E$

Si le flux est constant et en négligeant R_M, on a alors : $\overline{v} = \alpha . U = E = N . n . \Phi = k . n$

$$n = \frac{U}{k} . \alpha$$

D'où l'intérêt du hacheur : on peut faire varier la vitesse de rotation du moteur en réglant le rapport cyclique α

Le modèle électrique complet du moteur, de la bobine de lissage raccordés au hacheur est représenté par la figure ci-contre.



10- Exemples d'oscillateurs de commande du hacheur série :

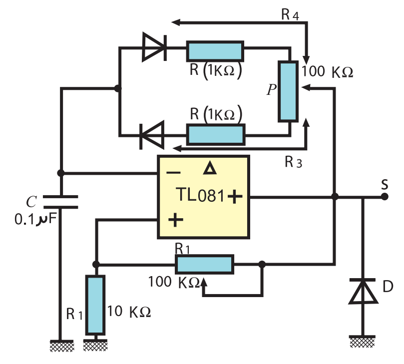
Pour commander le commutateur du hacheur série, il faut réaliser un montage électronique délivrant un signal en créneaux avec un rapport cyclique réglable. Il s'agit d'un oscillateur. Il existe plusieurs circuits intégrés réalisant cette fonction.

Exemple 1 :

Le courant de sortie de l'AOP étant très faible (10 à 20 mA), il conviendra d'utiliser un transistor Darlington derrière la sortie s.

Le rapport cyclique α est donnée par le rapport suivant :

$$\alpha = \frac{R_3}{R_3 + R_4}$$



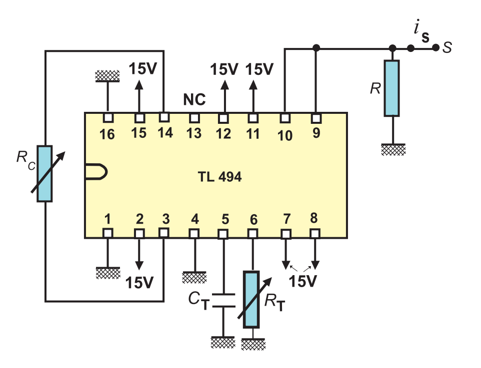
Exemple 2 :

(C_T = 0,33 μF, R_C = 10 kΩ, R_T = 22 kΩ, R = 1 kΩ/0,5W.

R_C règle le rapport cyclique.

R_T règle la fréquence.

$$f = \frac{1,1}{R_T . C_T}$$

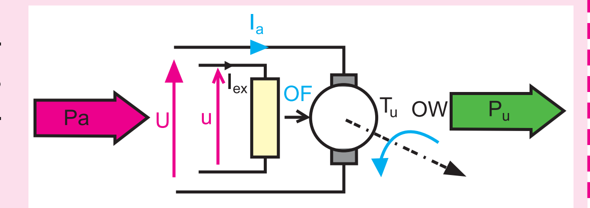


E- RESUMÉ :

Un moteur à courant continu se distingue par :

- la possibilité qu'il offre d'être **embarqué** dans des systèmes en déplacement ce qui est impossible avec les moteurs à courant alternatif qui doivent être reliés en permanence au réseau.
- son couple de démarrage très important puisque **Td**=K**Id**, avec **Id=U/R** et R de valeur **très faible.** Un moteur à courant continu peut alors démarrer en charge.

La figure ci-contre illustre le bilan des puissances et le tableau donné ci-dessous résume le bilan énergétique du moteur à courant continu :



$\eta = \frac{Pu}{Pa}$	Puissance absorbée totale	$Pa = U.Ia + u.i_{ex}$
	Puissance électromagnétique	$Pem = Pa - p_{ja} - p_{jex} = E.Ia = Tem.2.\pi.n$
	Pertes joules inducteur	$p_{jex} = u.I_{ex} = ri_{ex}^{2}$
	Pertes joules induit	$p_{ja} = RIa_a^2$
	Pertes collective	p_{cte} = est déterminée par un essai à vide
	Puissance utile	$Pu = Tu.\Omega$

Un moteur à courant continu peut être commandé en **vitesse** grâce à un **variateur** construit autour d'un **hacheur** de **rapport cyclique** α variable.

La trottinette électrique contribue très efficacement à la diminution de la pollution urbaine car elle n'émet aucun polluant à l'usage. Lorsque l'on parle de véhicule électrique, on imagine toujours un objet au coût exorbitant, dont les batteries remplissent le coffre et dont l'autonomie a du mal à dépasser les 100 kms. Mais si la voiture électrique peine à se sortir de cette image peut flatteuse, **les deux roues** arrivent à **maturité** et proposent de **vraies solutions** à la **pollution atmosphérique** et aux **problèmes de circulation**. Le constructeur des deux roues sort du lot en proposant (**enfin!**) des véhicules performants, à la **maintenance quasi nulle** et au **coût raisonnable**. Une trottinette a un fonctionnement équivalent à un celui d'un scooter 50 cm³ thermique, pour un prix d'achat beaucoup plus bas, et un prix de revient à l'usage beaucoup plus intéressant.



F- EXERCICES D'APPLICATION

1- Exercice corrigé :

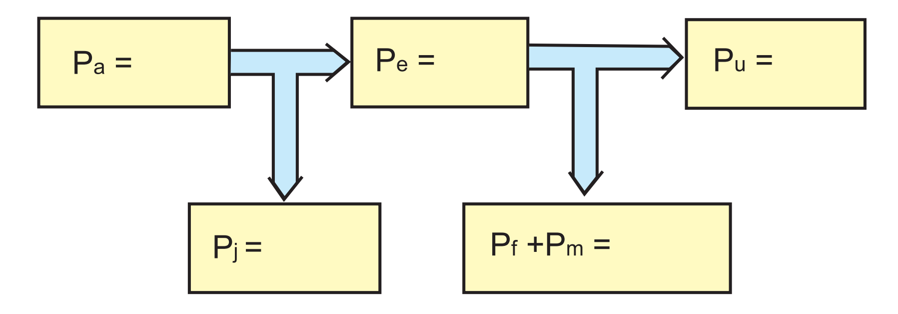
Un moteur à courant continu à excitation indépendante fonctionne à flux constant ; son courant inducteur a une intensité i_e = 0,35 A. Dans ces conditions, la force électromotrice E peut s'exprimer sous la forme E = k.n, relation dans laquelle n désigne la fréquence de rotation exprimée en tr/min ; on donne k = 0,11 $V/tr.min^{-1}$. La résistance de l'induit, mesurée à chaud est R= 6,3 Ω.

Fonctionnement à vide : sous tension d'induit nominal U = 250 V, l'induit absorbe un courant d'intensité I_0 = 0,28 A.

- Calculer la force électromotrice E_0 de l'induit dans ces conditions.
- En déduire la fréquence de rotation n_0 du moteur.
- Evaluer les pertes par effet Joule dans l'induit, notées p_{J0}.
- Déterminer le moment T_p du couple de pertes que l'on considérera constant dans la suite du problème.

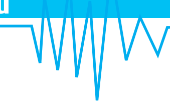
Fonctionnement en charge : le moteur, toujours alimenté sous tension nominale U=250 V, développe un couple électromagnétique de moment T_e = 2,1 N.m.

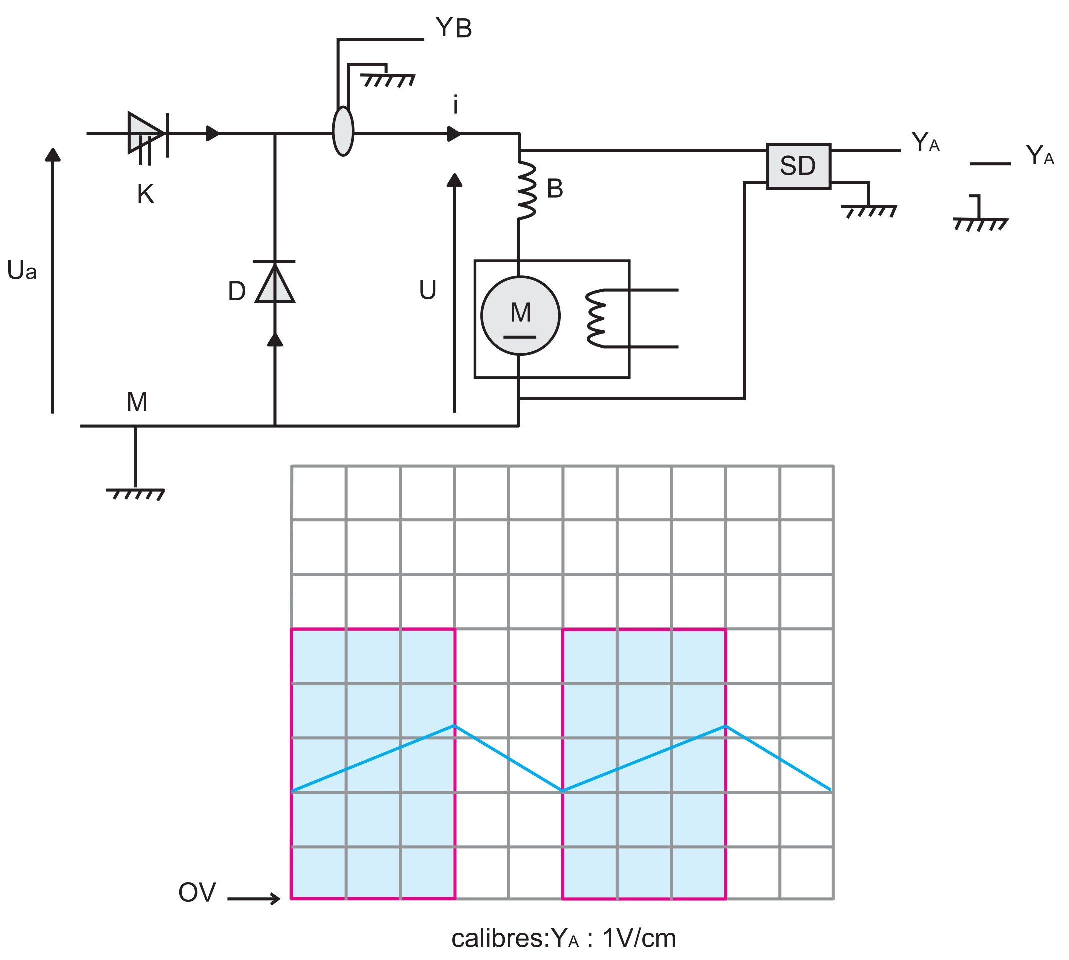
- Montrer que l'induit absorbe alors un courant d'intensité I = 2,0 A.
- Calculer la force électromotrice E du moteur.
- En déduire sa fréquence de rotation, n.
- Le schéma ci-dessous, représente le bilan des puissances de l'induit en charge; en justifiant les calculs effectués, compléter ce schéma en donnant la valeur des différentes puissances mises en jeu.
- Calculer le rendement de l'induit du moteur en charge.
- Calculer le moment Tu du couple utile développé par le moteur.



Etude du hacheur :

L'induit du moteur étudié est alimenté par un hacheur série dont le schéma est représenté ci-dessous. Les interrupteurs électroniques utilisés sont supposés parfaits. Une bobine de lissage, B, de résistance négligeable, est placée en série avec l'induit ; la résistance de celui-ci est il est caractérisé par la relation E = k.n dans laquelle E est exprimée en V et n en tr/min. On donne k = 0,11 $V/tr.min^{-1}$.





calibres: Y_A : 1V/cm

Y_B : 50 mV/cm

0,2 ms/cm

Commande de l'interrupteur K ; T désigne la période de fonctionnement :

$0 < t < \alpha T$: K fermé.

$\alpha T < t < T$: K ouvert.

En charge, pour un certain régime de fonctionnement, on a relevé à l'oscilloscope les variations de u et i conformément au schéma de la figure ci-dessus. Les oscillogrammes obtenus sont représentés sur la même figure. Pour relever ces oscillogrammes on a utilisé une sonde de tension de rapport 1/50 et une sonde de courant de sensibilité 100mV/A.

-Déterminer la fréquence de fonctionnement f du hacheur.

-Quelle est la valeur du rapport cyclique α du régime étudié ?

-$< u >$ désigne la valeur moyenne de la tension aux bornes de l'ensemble moteur + bobine de lissage ; montrer que $< u > = 150$ V.

-Déterminer la valeur maximale I_M et la valeur minimale, I_m, de l'intensité du courant absorbé par l'induit du moteur.

-En déduire l'ondulation $\Delta i = I_M - I_m$ du courant et sa valeur moyenne $< i >$.

-Ecrire la relation entre $< u >$, $< i >$, n, R et k.

-Calculer la fréquence de rotation n du moteur.

Corrigé

Fonctionnement à vide :

$U=E_0 + R_{10}$ d'où $E_0 = U - R_{10}$= 250 - 6,3 x 0,28 = 248 V.
$E_0 = k\, n_0$ d'où $n_0 = E_0 / k$ = 248/0,11= 2260 tr/min.
pertes par effet joule dans l'induit : $P_{j0} = RI_0^2$= 6,3 x $0,28^2$ = 0,5 W
puissance absorbée = pertes joule + pertes fer + pertes mécaniques
Pa = U_{10}= $P_{J0} + P_f + P_m$ d'où $P_f + P_m = U_{10} - P_{J0}$ = 250x0,28-0,5 = 69,5 W.
Couple de pertes : il correspond aux pertes ferromagnétiques et aux pertes mécaniques.
$\Omega_0 = 2\pi\, n_0/60$ = 6,28 x 2260/60 = 236,5 rad/s
couple de pertes : $T_p = (P_m + P_f) / \Omega_0$ = 69,5 / 236,5 = 0,294 Nm.

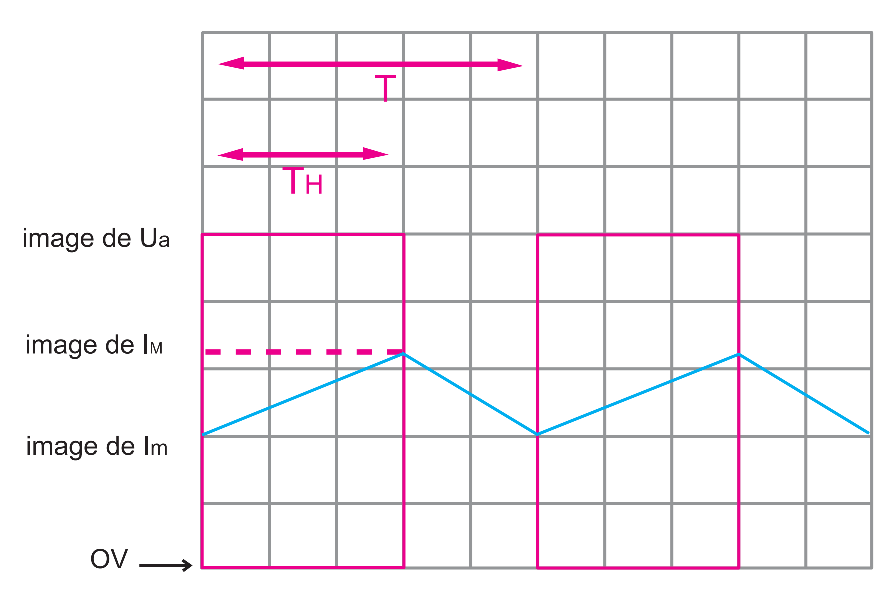
Fonctionnement en charge :

Relations fondamentales du moteur à courant continu.
$E= K\,\Omega$ = kn avec $\Omega = 2\pi.n/60$ d'où $E= K2\pi\, n/60$
soit K = 60k / (2p)=60*0,11/6,28=1,05
couple électromagnétique : Te = K I = 60k / (2π) I d'où I = 2π Te/(60k)
I = 6,28 x 2,11 / (60*0,11) = 2A.
E = U - RI = 250 - 6,3 x 2 = 237,4 V.
fréquence de rotation : n = E/k = 237,4 /0,11 = 2158,2 tr/min.
Calcul des diverses puissances.
puissance absorbée : Pa = UI = 250 x 2 = 500 W
pertes joule : Pj = RI^2 = $6,3x2^2$ = 25,2 W

Deux façons de calculer la puissance électromagnétique :

Pe = Pa - P_j = 500 - 25,2 = 475 W ;
Pe = $Te\Omega$ =Te 2π n/60 = 2,11x6,28x2158,2 / 60 = 476,6 W
$P_m + P_f = Tp\Omega = Tp2\pi\, n/60$ =0,294 x 6,28 x158,2/60 = 66,4 W
Pu = Pe - (Pf + Pm) = 476,6 - 66,4 = 410,2W.
Rendement de l'induit : η=Pu/Pa= 410,2 / 500 = 0,82 (82%)
Moment du couple utile :
Tu = Pu / Ω =Pu / (2π n) x 60 = 1,815 Nm.

Etude du hacheur :



calibres: Y_A : 1V/cm

Y_B : 50 mV/cm

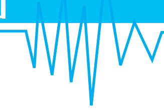
0,2 ms/cm

Sur l'oscillogramme, on observe un signal rectangulaire qui correspond à la tension hachée u, et un signal triangulaire correspondant au courant i.
Leurs périodes s'étalent sur 5 carreaux, d'où une période :
T = 5 x base de temps = 5 x 0,2 =1ms = 10^{-3} s.
et une fréquence de fonctionnement du hacheur : 1 / 10^{-3} = 1000 Hz.
Sur ce même oscillogramme, la durée à l'état haut de la tension u s'étale sur 3 carreaux, comme les durées sont proportionnelles aux longueurs mesurées sur l'oscillogramme, on a :

$$\alpha = TH/T = 3 / 5 = 0{,}6.$$

Or, sur l'oscillogramme, l'amplitude de l'image de u (Û=Ua) mesure 5 carreaux soit 5 x1 = 5V.
On utilise une sonde de tension 1/50, d'où : 5x50 =250 V.
<u>= 0,6 x 250 = 150 V.

Le signal triangulaire correspond au courant i,
On mesure :
- Valeur maximale : 3,2 carreaux soit une tension égale à 3,2 x 50 = 160mV.
On utilise une sonde de courant de 100mV / A d'où I_M = 1,6 A
- Valeur minimale : 2 carreaux : 2x50 = 100mV soit I_m = 1A .
Ondulation : Δi = 1,6 -1 = 0,6 A.



- Valeur moyenne d'un signal triangulaire : $<i>= (I_M+I_m)/2 = (1,6+1)/2 = 1,3$ A.
La loi des mailles permet d'écrire : $u = u_M+u_L$
(u_M est la tension aux bornes du moteur et u_L celle aux bornes de l'inductance).
En valeurs moyennes, on obtient : $<u >= <u_M>+<u_L>$
or la tension moyenne aux bornes de l'inductance est nulle d'où :
$<u >= <u_M> = E + R<i>=k.n+ R<i>$
$n = (<u >-R<i>) /k = (150-6,3*1,3) /0,11 = 1290$ tr/min.

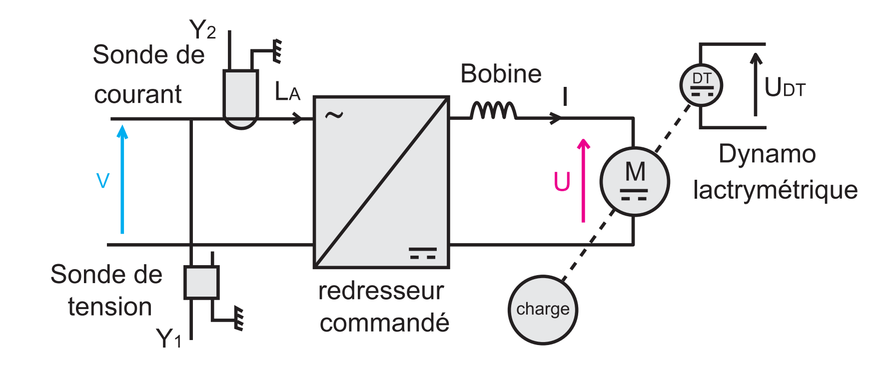
2- Exercice à résoudre :

On étudie un moteur à courant continu, son alimentation et le système qui assure la régulation de sa vitesse.
L'intensité d'excitation du moteur est maintenue constante. Le constructeur donne le point de fonctionnement suivant :
- tension d'induit : U = 260 V ;
- intensité d'induit : I = 17,5 A ;
- fréquence de rotation : n = 1600 tr/min.

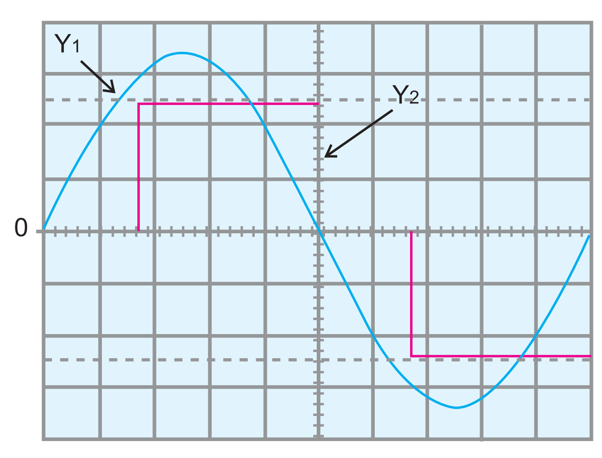
La résistance de l'induit est R = 1,73 W .

1- Calculer la force électromotrice (f.é.m.) E pour le fonctionnement donné.
2- A intensité du courant d'excitation constante, la f.é.m. E est proportionnelle à la fréquence de rotation. Montrer que E = 0,144.n (avec n en tr/min).
3- Déterminer la fréquence de rotation pour U = 160 V et I = 12,0 A.
La tension U gardant la valeur de 160 V, l'intensité I du courant traversant l'induit devient égale à 17,5 A à la suite d'une augmentation de charge du moteur.
4- Calculer la nouvelle valeur de la fréquence de rotation n.
5- Après cette augmentation de l'intensité I, comment faut-il agir sur la tension U pour retrouver la valeur initiale de n ?

Le circuit d'alimentation (redresseur commandé) de l'induit est représenté par le schéma ci-contre :
Grâce à la bobine, l'intensité I et la tension U de l'induit sont constantes. On visualise les grandeurs v et i_A sur un oscilloscope en utilisant une sonde de tension différentielle de rapport 1/200 et une pince de courant de sensibilité 100 mV/A.



L'oscillogramme obtenu est donné sur la figure ci-dessous :

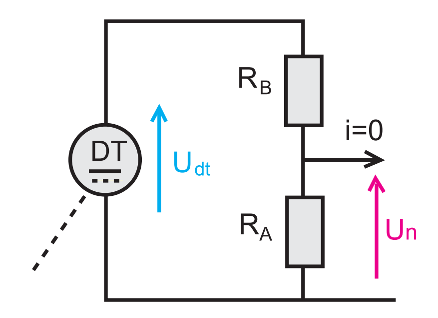


Y1 : 0,5 V/div Y2 : 0,5 V/div

Base de temps : 2 ms/div

5- Le redresseur commandé est-il un pont tout thyristor ou un pont mixte ? Justifier en s'appuyant sur l'une des courbes de l'oscillogramme.
6- Déterminer la valeur efficace V de la tension v.
7- Déterminer la valeur maximale îA de l'intensité iA et en déduire que l'intensité I est égale à 12 A.

Le moteur entraîne une dynamo tachymétrique qui est munie de deux résistances. Quand la fréquence de rotation n est égale à 1 000 tr/min, la tension u_{DT} vaut 60V.



8- Quel est le rôle de la dynamo tachymétrique ?
9- Donner l'expression de la tension un en fonction de uDT, RA et RB.
10- RA = 10,0 kW. Calculer RB de façon à obtenir un égale à 10,0 V quand la fréquence de rotation vaut 1 000 tr/min.

Electronique

C1 Les amplificateurs linéaires intégrés

Partie Electronique

LES AMPLIFICATEURS LINÉAIRES INTÉGRÉS A.L.I

A- MISE EN SITUATION

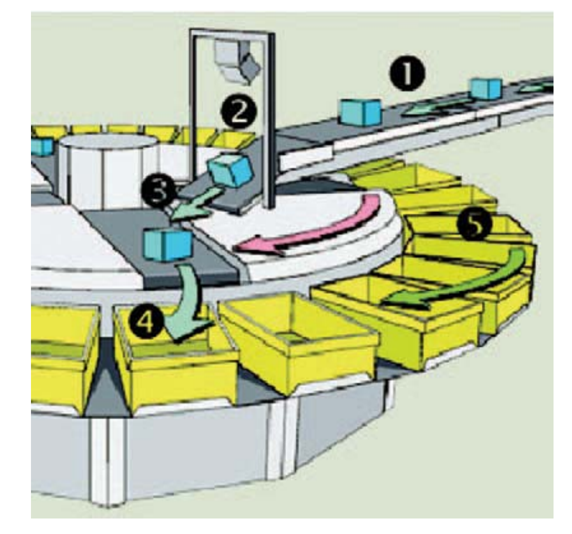
1- Présentation du système : Système de tri avec alimentation et déchargement automatique des postes de destination



Le véritable **tri des pièces** s'effectue a l'aide de bandes disposées en étoile tournante. A sa circonférence, le système est équipe d'une couronne tournante entraînée séparément sur laquelle s'animent des bacs de destination.

En cours de tri, des commandes sont alors complétées à des heures différentes. Sur la station d'échange automatique, les bacs avec commandes complètes sont transférés. Puis un bac vide est mis automatiquement à la place vacante, destiné logiquement a une nouvelle commande.

Les pièces triées sont identifiées à l'accès dans la zone de tri. Le code à barres est un standard international de **codification des biens**. Chaque article a un code qui lui est propre représenté par un ensemble de barres foncées sur fond clair. Ce code est lisible automatiquement par des appareils de lecture basé sur des capteurs optiques et des amplificateurs appelés **lecteurs de code à barres**.



1=Alimentation
2=Enregistrement du code à barres
3=Transfert vers bandes en étoile tournante
4=Dépose au poste de destination
5=Couronne tournante



2- Problèmes posés :

a) Qu'est ce qu'un comparateur ?
b) Quelle est la notion de seuils ?
c) Quelles sont les applications de ces comparateurs ?

B- RAPPEL :

Activité

Réaliser l'activité N°1 du TP-C1 dans le manuel d'activités

Il s'agit, pour un amplificateur linéaire, d'identifier sur le schéma structurel de la carte de commande du système de tri SET :

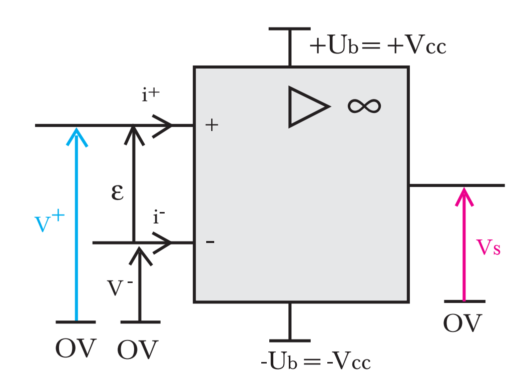
- le régime de fonctionnement
- la fonction qu'il assure.

1- Présentation de l'amplificateur linéaire intégré :

L'amplificateur opérationnel le plus couramment utilisé est un amplificateur de différence à référence commune. Il possède :

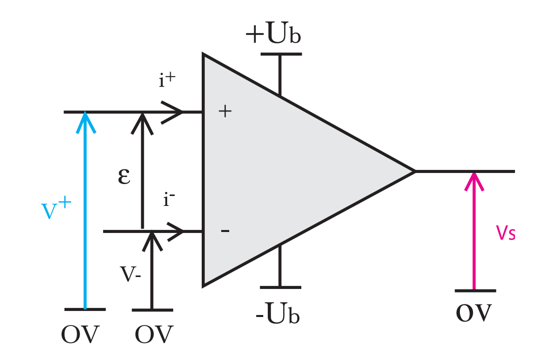
- **Deux entrées** : Une **entrée** notée (**–**) dite entrée **inverseuse** et une **entrée** notée (**+**) dite entrée **non inverseuse**.
- **Une sortie** : **Vs**.

Les symboles sont représentés ci-dessous.





Symbole NF C 03-213



Symbole Américain

Le composant comporte de plus, des broches d'**alimentation (Ex : +Ub et –Ub)** et est souvent alimenté de manière **symétrique** par rapport au **0V de référence (Ex : +15V, –15V).**

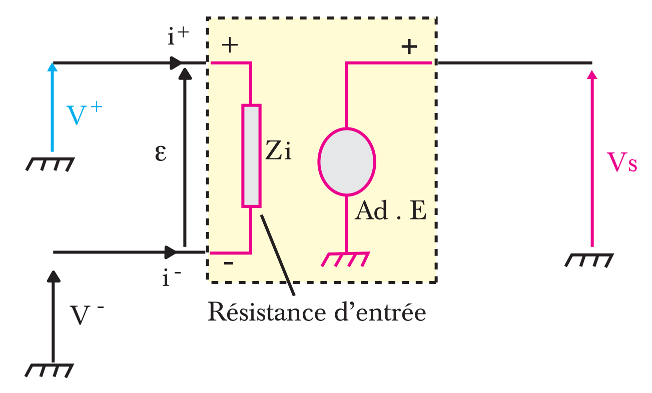
Les potentiels des entrées sont repérées par rapport à la référence commune de toutes les tensions (**masse ou 0V de référence**), et sont nommés V^+ et V^-.

La différence de potentiel entre **l'entrée +** et **l'entrée –** est appelée **tension d'entrée différentielle** et est notée **Vd** (ou parfois ε) :

$$Vd = (V^+ - V^-).$$

2- Caractéristiques d'un ALI idéal :

- L'amplification différentielle (**Ad = Vs/Vd**) est infinie.
- La tension différentielle d'entrée **(Vd = V^+ – V^-)** est nulle
- L'impédance d'entrée différentielle **Ze** est infinie
- L'impédance de sortie **Zs** est nulle
- Les intensités des courants d'entrée **i^+** et **i^-** sont nulles.

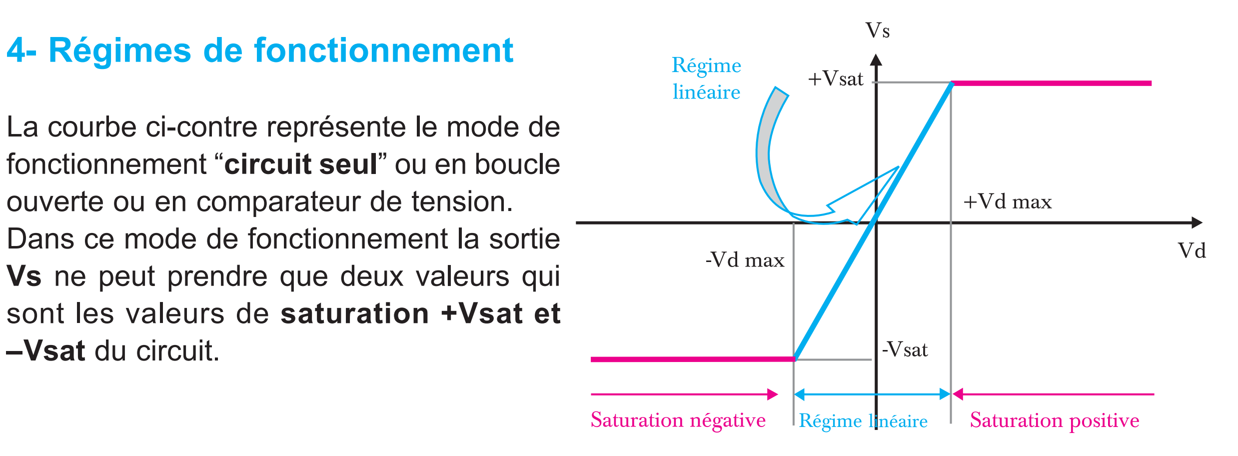


3- Fonctionnement d'un amplificateur :

Un amplificateur linéaire intégré peut fonctionner en :
- amplificateur de signaux faibles selon un gain préalablement calculé.
- comparateur d'un signal par rapport à une référence donnée.
- trigger de Schmidt.
- générateur de signaux carré et triangulaire.
- additionneur, soustracteur de divers signaux.

4- Régimes de fonctionnement

La courbe ci-contre représente le mode de fonctionnement "**circuit seul**" ou en boucle ouverte ou en comparateur de tension. Dans ce mode de fonctionnement la sortie **Vs** ne peut prendre que deux valeurs qui sont les valeurs de **saturation +Vsat et –Vsat** du circuit.



C- COMPARATEUR SIMPLE SEUIL

Activité

Réaliser l'activité N°2 du TP-C1 dans le manuel d'activités

Il s'agit pour amplificateur linéaire fonctionnant en boucle ouverte de tracer l'allure de la tension de sortie en fonction des entrées :
- pour une polarisation symétrique
- pour une polarisation asymétrique.

En boucle ouverte la présence de la moindre tension ε porte la sortie en **saturation** (du fait de la grande valeur de Ad).

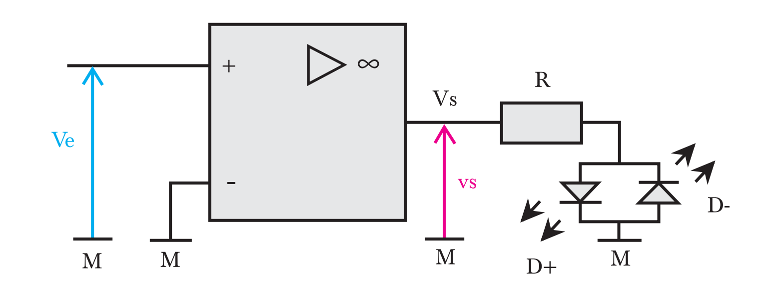
NB : On ne peut pas poser **ε = 0**. Cette condition n'est possible que lors des transitions de la sortie (passage de **vs** de **+Vsat** à **–Vsat** ou l'inverse).

- Si **ε > 0**, **Vs = +Vsat** = (+Vcc - tension de déchet à l'état haut)
- Si **ε < 0**, **Vs = –Vsat** = (–Vcc + tension de déchet à l'état bas)

Exemple 1 :

Détection de signe.Ve = ε

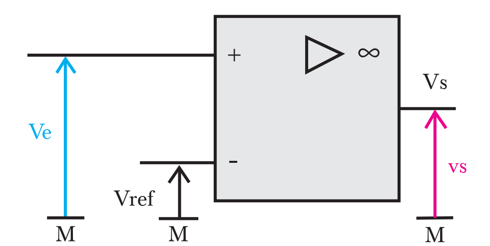
- Si **Ve > 0**, **Vs = +Vsat**, donc **D +** est **allumée**.
- Si **Ve < 0**, **Vs = –Vsat** et **D –** est **allumée**.

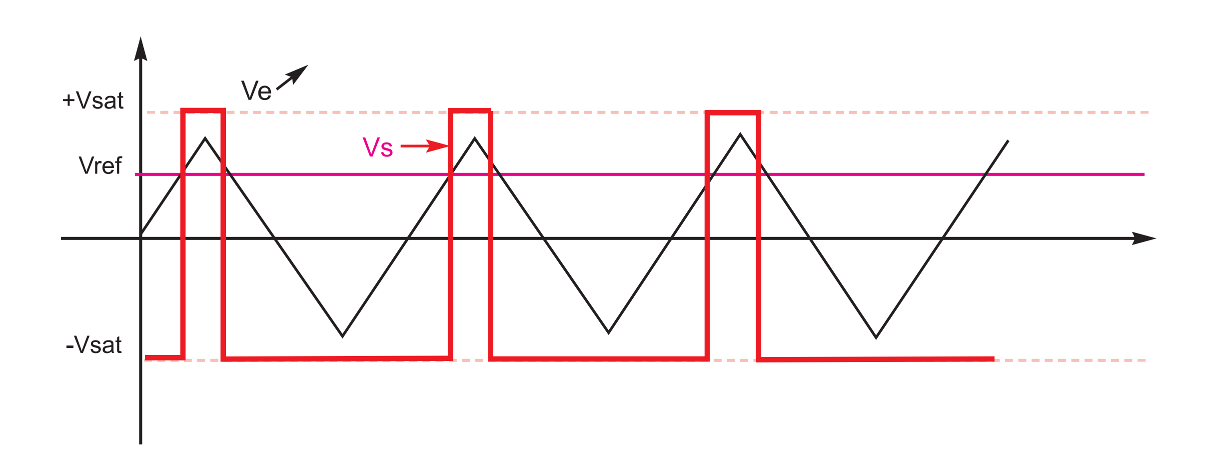


Exemple 2 :

Comparateur de tensions

- Si **Ve > Vref, Vs = +Vsat**
- Si **Ve < Vref, Vs = –Vsat**





Remarque :
Les comparateurs simples sont sensibles au bruit. En effet, lorsque la tension d'entrée UE est voisine de la tension de référence, une tension de bruit peut provoquer intempestivement et aléatoirement le basculement du comparateur.

C- COMPARATEUR A DOUBLE SEUILS :

Activité

Réaliser l'activité N°3 du TP-C1 dans le manuel d'activités

Il s'agit d'identifier des amplificateurs linéaires fonctionnant en :

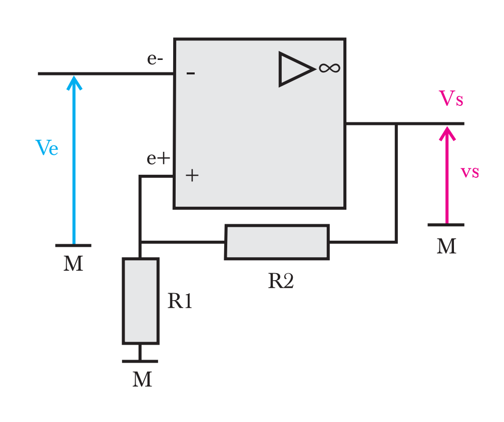
- en boucle ouverte
- en boucle fermée

1- Comparateur inverseur à double seuils

Dans ce cas (comme dans le cas de la boucle ouverte), la tension **ε** n'est **pas nulle** (sauf au moment de la commutation de Vs). La sortie est portée en saturation comme dans le cas du comparateur. La sortie est rebouclée vers l'entrée **+,** et provoque donc deux seuils de comparaison (suivant que **Vs = +Vsat** ou **Vs = –Vsat**).

Ve^{+} = (R1/(R1+R2))xVs et Ve^{-} = Ve
d'où

$$\varepsilon = V_e^+ - V_e^- = \frac{R1.Vs}{(R1+R2)} - Ve$$



avec Vs = +Vsat ou –Vsat suivant le signe de **ε**

- **ε > 0, Vs= +Vsat**

donc **Ve < (R1/R1+R2) x Vsat**

- **ε < 0, Vs= –Vsat**

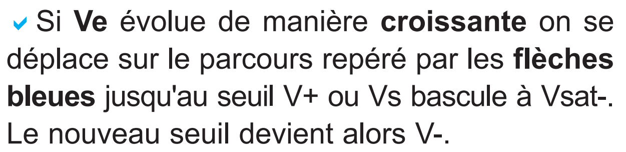
donc **Ve > – (R1/R1+R2)xVsat**

Il s'agit d'un comparateur à 2 seuils symétriques V^{+} et V^{-}, dont la caractéristique de transfert est donnée ci-contre :

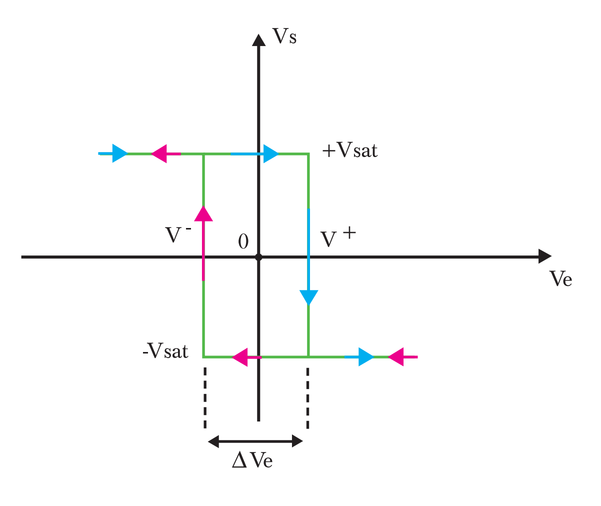
V^{+} = (R1/R1+R2) x Vsat

et

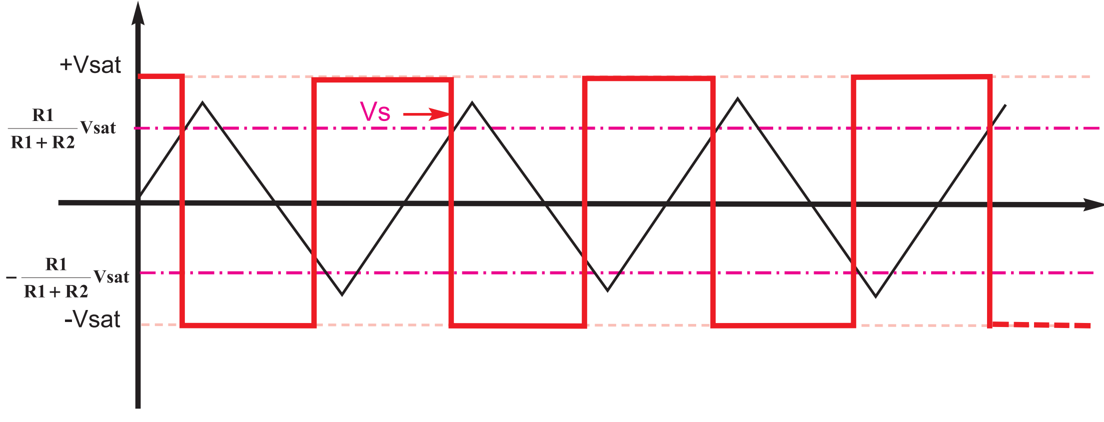
V^{-} = –(R1/R1+R2)x Vsat



- Si **Ve** évolue de manière **croissante** on se déplace sur le parcours repéré par les **flèches bleues** jusqu'au seuil V+ ou Vs bascule à Vsat-. Le nouveau seuil devient alors V-.
- Si **Ve** évolue ensuite de manière **décroissante** on se déplace sur le parcours repéré par les **flèches rouges**.



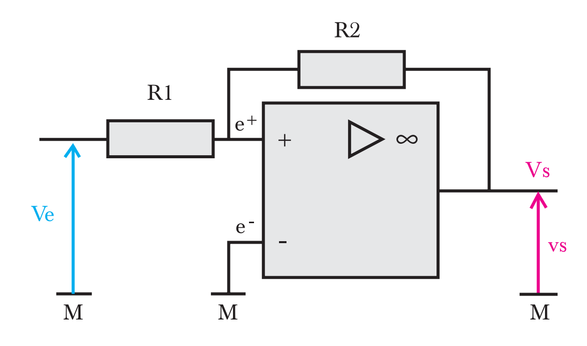


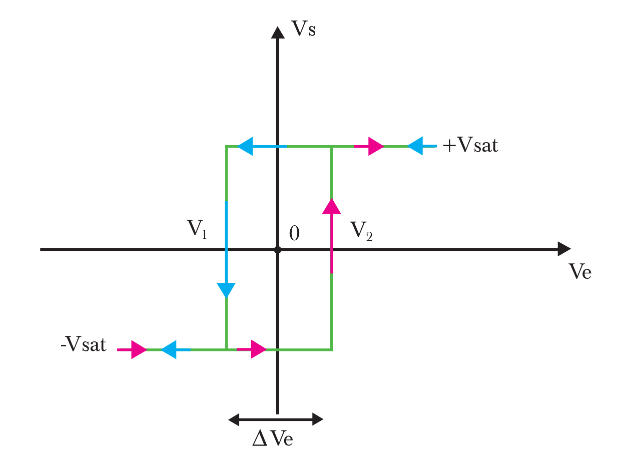


2- Comparateur non inverseur à double seuils :

Le basculement aura lieu pour ε = 0. Donc
pour **Ve = - (R1/R2)xVs**
avec Vs = +Vsat ou -Vsat
-Vcc < Ve < +Vcc impose R1 < R2
afin que Ve puisse atteindre les seuils.

$$\varepsilon = \frac{R1.Vs + R2.Ve}{(R1 + R2)}$$





On obtient la caractéristique de transfert ci-contre :

$$V_2 = (R1/R2)xVsat$$
et
$$V_1 = -(R1/R2)xVsat$$

D- FONCTIONNEMENT EN REGIME LINEAIRE :

Activité

Réaliser l'activité N°3 du TP-C1 dans le manuel d'activités

Il s'agit de tracer la courbe de sortie pour un amplificateur linéaire bouclé :

- sur un circuit résistif avec un double circuit d'attaque résistif
- sur un circuit capacitif avec un circuit d'attaque résistif
- sur un circuit résistif avec un circuit d'attaque capacitif.

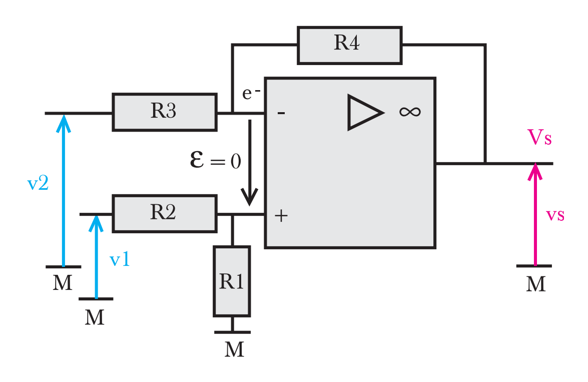


1- Fonctionnement en amplificateur de différence :

Il s'agit d'une configuration en amplificateur non inverseur pour e^+ et inverseur pour e^- d'où :

$$Ve^+ = \frac{V1.R1}{(R1+R2)}$$

$$\text{et} \quad Ve^- = \frac{Vs.R3+V2.R4}{(R1+R2)}$$



or $e^+ = e^-$ d'où

$$Vs = [(1+\frac{R4}{R3}).\frac{R1}{(R1+R2)}].V1 - \frac{R4}{R3}.V2$$

$$\text{Si } R3/R4 = R2/R1 \Rightarrow Vs = R4/R3.(V1-V2)$$

$$\text{Si de plus } R1 = R2 \text{ et } R3 = R4 \Rightarrow Vs = V1 - V2$$

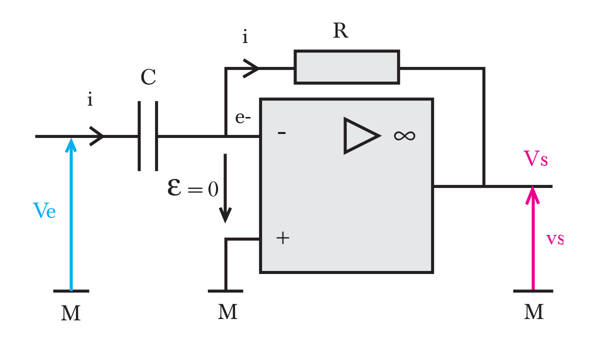
2- Fonctionnement en dérivateur :

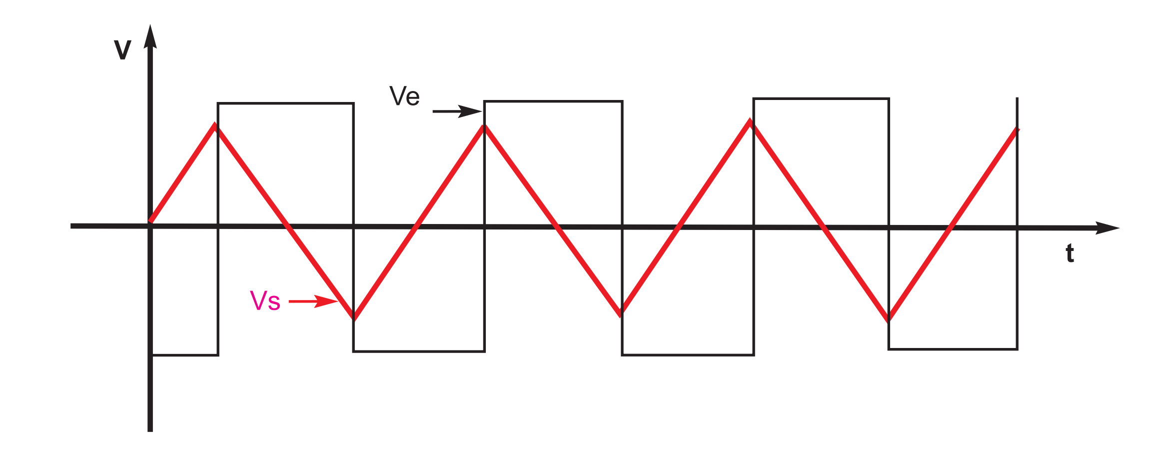
$$i = C.(\frac{dVe}{dt}) \text{ et } i = -\frac{Vs}{R}$$

$$\text{d'où } Vs = -RC.(\frac{dVe}{dt})$$

La sortie est proportionnelle à la dérivée de l'entrée. Le coefficient de proportionnalité est égal à la constante de temps RC.

NB : Ce montage ne fonctionne qu'avec une résistance r (r de l'ordre de 100Ω) en série avec C. Sans r, il se produit des oscillations.

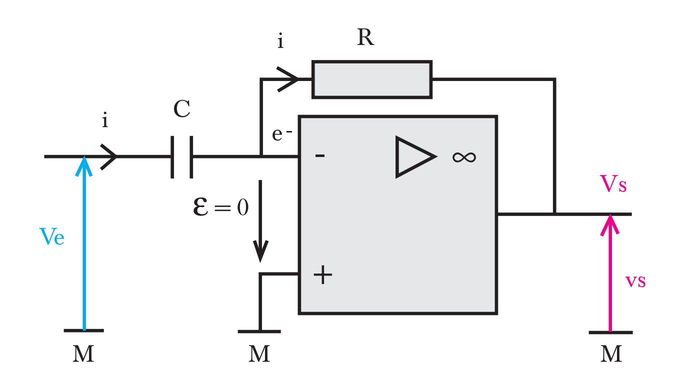




3- Fonctionnement en intégrateur :

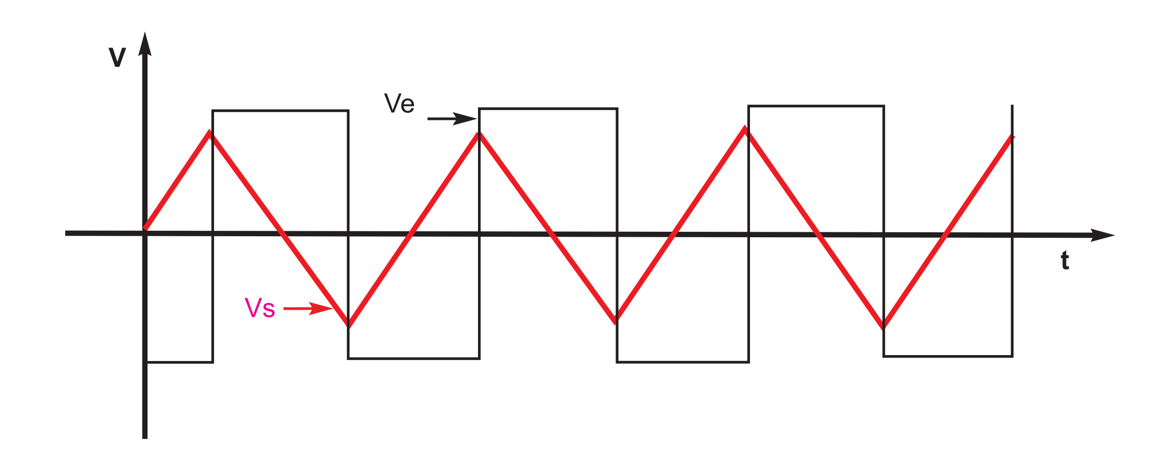
$$i = -C.(\frac{dVs}{dt}) \text{ et } i = -\frac{Ve}{R}$$

$$\text{d'où } Ve = -RC.(\frac{dVs}{dt})$$



Donc la sortie vs est proportionnelle à la primitive de l'entrée.

$$\text{donc} \quad Vs = -\frac{1}{RC} \int Ve.dt$$



E- APPLICATION DES AMPLIFICATEURS LINEAIRES INTEGRES :

1- Conversion

Activité

Réaliser l'activité N°3 du TP-C1 dans le manuel d'activités

Il s'agit de simuler, sur logiciel ou sur maquette d'étude, le fonctionnement d'un convertisseur :

- numérique/analogique à resistances pondérées
- analogique/numérique simple rampe.

Les systèmes numériques n'admettent en entrées que des grandeurs logiques et ne peuvent délivrer que des sorties de la même nature. Il est alors indispensable de leur associer :

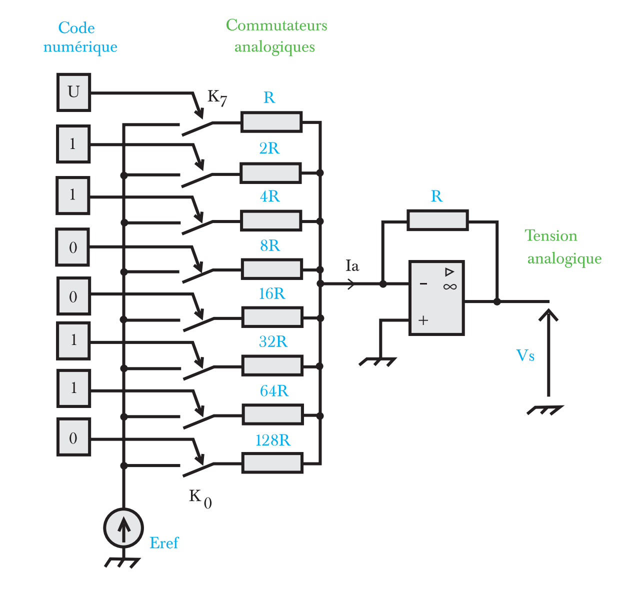
- à l'entrées des convertisseurs **analogiques/numériques (CAN)** pour traduire les signaux issus de divers capteurs en informations exploitables.
- à la sortie des convertisseurs **numériques/analogiques (CNA)** pour permettre de piloter des actionneurs électriques ou autres.



2- Convertisseur Numérique/ Analogique à résistances pondérées

Pour convertir une information numérique en un signal analogique, plusieurs procédés sont utilisés. Il s'agit d'utiliser un sommateur inverseur avec en entrées des résistances pondérées.

- une tension de référence Eref,
- une batterie de commutateurs commandés par le code numérique contenu dans les cases d'un registre binaire,
- une batterie de résistance pondérées de manière à ce que les courants générés soient dans une progression géométrique de raison 1/2.



Dans l'exemple de la figure 1, le signal analogique de sortie est la somme des courants collectés par les commutateurs fermés. L'amplificateur opérationnel peut être utilisé pour traduire ce signal en tension.
On obtient par exemple, pour l'information numérique N = 01100110, un courant analogique égal à :

$$Ia = Eref\left(\frac{1}{2R} + \frac{1}{4R} + \frac{1}{32R} + \frac{1}{64R}\right)$$

D'une manière générale : $Ia = Eref\left(\frac{k_7}{R} + \frac{k_6}{2R} + \frac{k_5}{4R} + \frac{k_4}{8R} + \frac{k_3}{16R} + \frac{k_2}{32R} + \frac{k_1}{64R} + \frac{k_0}{128R}\right)$

Les différents Ki prenant la valeur 0 ou 1, Ia peut aussi s'écrire :

$$Ia = \frac{Eref}{128R}(k_0.2^0 + k_1.2^1 + k_2.2^2 + k_3.2^3 + k_4.2^4 + k_5.2^5 + k_6.2^6 + k_7.2^7)$$

Pour ce type de convertisseur, l'amplificateur opérationnel transforme le courant en une tension. L'amplificateur opérationnel fonctionne en régime linéaire (contre-réaction sur l'entrée e^- de l'ALI). Le potentiel sur l'entrée e^- est égal au potentiel de l'entrée e^+, on obtient **$\varepsilon = Ve^+ - Ve^- = 0V$**. Dans ces conditions :

$$Vs = -R.Ia = -\frac{Eref}{128}(k_0.2^0 + k_1.2^1 + k_2.2^2 + k_3.2^3 + k_4.2^4 + k_5.2^5 + k_6.2^6 + k_7.2^7)$$

F- RESUMÉ

Comparateur de tension : Un comparateur de tensions est un composant à deux entrées et une sortie, dont la fonction est de fournir une tension de sortie vs fonction du **signe** de la **tension** différentielle d'entrée **Vd.**

Comparateur simple : Un simple est réalisé à l'aide d'un ALI en boucle ouverte. La tension de référence Vref est appliquée sur l'une des entrées de l'ALI et Ve sur l(autre. Il existe deux types de comparateurs simples selon que le signal d'entrée est appliqué sur l'entrée non inverseuse ou sur l'entrée inverseuse.

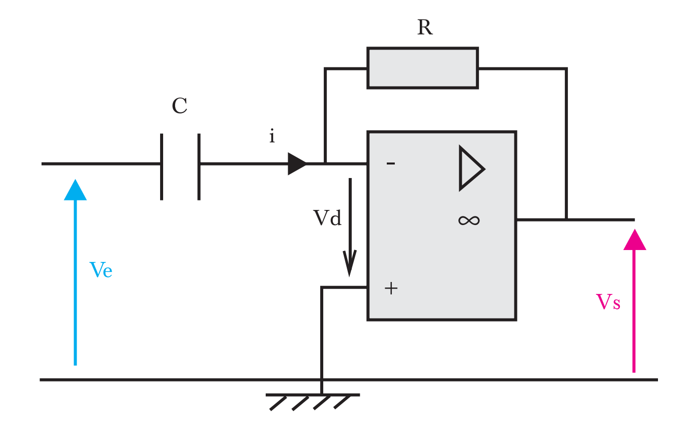
Comparateur à double seuils (à hystérésis) : Un comparateur à hystérésis est réalisé à l'aide d'un ALI possédant une boucle de réaction aboutissant sur l'entrée non inverseuse. Il existe deux types de comparateurs à hystérésis que le signal d'entrée Ve est appliqué non inverseuse ou sur l'entrée inverseuse. Il est utilisé dans la conception des multivibrateurs astables. En effet le comparateur est associé à un circuit complémentaire dont le rôle est de faire glisser le point de fonctionnement vers les points de basculement.

Dans **le système de tri**, les différents capteurs génèrent des signaux de forme irrégulière. Pour les rendre exploitables, on utilise des circuits électroniques de mise en forme constitué essentiellement de **comparateurs** à base d'**amplificateurs linéaires intégrés** (Triggers de Schmidt).

G- EXERCICES D'APPLICATION

Exercice 1 :

On considère le montage suivant :

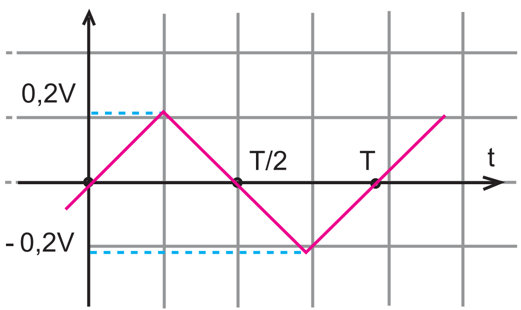


R=10kΩ , C=0,1µF



a) Calculer pour chaque montage, la fonction de transfert : T = Vs/Ve en considérant l'amplificateur opérationnel idéal.

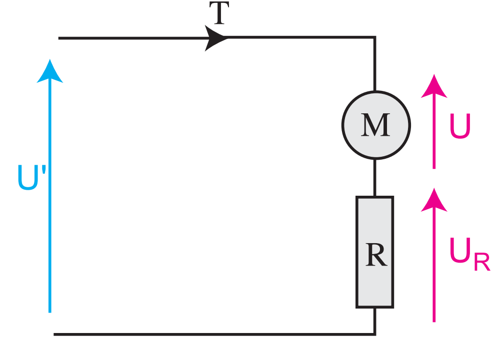
b) On applique à l'entrée du montage le signal de la figure ci-contre. Donner l'allure de la tension recueillie à la sortie du montage. Quelle est la fonction réalisée par ce montage.



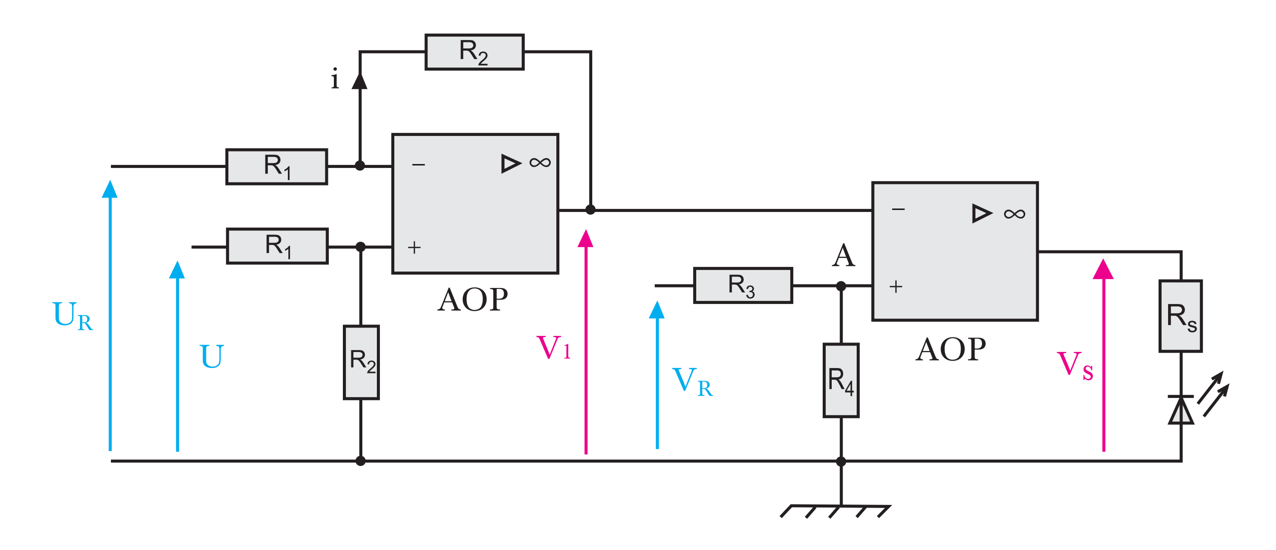
Exercice 2 :

Les amplificateurs opérationnels sont parfaits, ils sont polarisés par une alimentation symétrique + 15 V, –15 V. On admet que leurs tensions de saturation sont également de + 15 V, –5 V.

- **U** est la tension aux bornes de l'induit du moteur.
- **U'** est la tension aux bornes de l'ensemble moteur - résistance **R**.
- Le moteur possède une f.c.e.m **E** et une résistance interne $\mathbf{R_A}$.



Les tensions $\mathbf{U_R}$ et **U** sont appliquées à l'entrée du montage de la figure suivante :



R_3 = 4,7 kΩ ; R_4 = 5,2 kΩ ; R_s = 1 kΩ ; V_R = 10 V

Etude du premier amplificateur opérationnel :

1) Le fonctionnement est-il linéaire ?
2) Donner le potentiel V^+ en fonction de U.
3) Donner le potentiel V^- en fonction de U_R et V_1.
4) Donner l'expression de V1 en fonction de U et U_R., montrer que V1 peut se mettre sous la forme : V1 = k (U - UR). On donne :

- Les résistances R et RA sont égales à 0,2Ω.
- L'amplification en tension est k = 1 / 20
- La f.c.e.m est de la forme $E = 8{,}83.10^{-2}.n$ avec la f.e.m E en V et n en $tr.min^{-1}$.

5) Quel est l'intérêt de choisir les deux résistances R et R_A égales ?
6) Donner les tensions U et U_R en fonction de n et de l'intensité du courant I.
7) En déduire l'expression de V_1 en fonction de la fréquence de rotation.
8) Quel est l'intérêt de ce montage ?
9) Calculer V_1 à l'arrêt puis pour une fréquence de rotation de 1250 $tr.min^{-1}$.

Etude du second amplificateur opérationnel :

1) Le fonctionnement est-il linéaire ?
2) Calculer le valeur du potentiel au point A.
3) Si v_1 = 3 V, quelle est la valeur de vs ? Quel est l'état de la LED ?
4) Si v_1 = 6 V, quelle est la valeur de vs ? Quel est l'état de la LED?
5) Pour quelle valeur de la fréquence de rotation la LED change-t-elle d'état ?

Exercice 3 : Contrôle de la pression d'un pneu

On désire contrôler la pression de gonflage de pneus sur une chaîne de montage automobile afin de la situer par rapport à deux pressions limites :

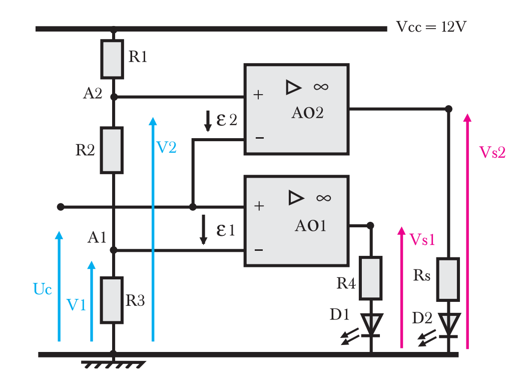
P_{inf} = 200 kPa P_{sup} = 260 kPa (Pa : Pascal)

La pression P dans les pneus est mesurée par un capteur de pression qui délivre une tension u_C telle que : $u_C = -(5 \times 10^{-5} \times P) + 19$. Dans cette formule u_C est exprimée en V et P en Pa.

La tension u_C est appliquée à l'entrée du montage ci-contre :

Les deux amplificateurs opérationnels AO1 et AO2 sont supposés parfaits. Ils sont alimentés sous tensions dissymétriques 0 V / 12 V (non représentées sur le schéma). Dans ce cas les tensions de saturation sont V_B=0V et V_H =+ 12V.

Les diodes électroluminescentes D_1 et D_2 présentent à leurs bornes une tension $V_{D1} = V_{D2}$ = 2 V lorsqu'elles sont passantes.



On donne : $R_2 = 5$ k Ω ; $R_3 = 10$ k Ω et V_{CC} = 12 V.



1) Calculer u_C pour les valeurs des pressions limites : soit u_{C1} pour Psup et u_{C2} pour Pinf.
2) Exprimer les tensions V1 et V2 en fonction de VCC, R1, R2 et R3.
3) Exprimer la tension différentielle ε_1 en fonction de u_C et V_1 et la tension différentielle ε_2 en fonction de u_C et V_2.

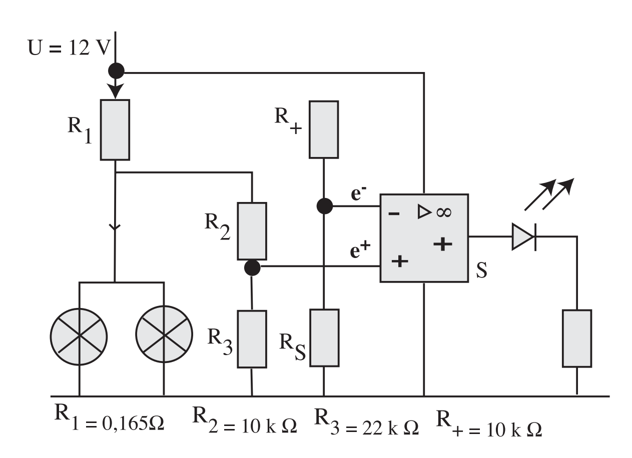
Les amplificateurs AO1 et AO2 fonctionnent en régime non linéaire.
1) En déduire les valeurs :
de V_{S1} pour $\varepsilon_1>0$ et $\varepsilon_1<0$;
de V_{S2} pour $\varepsilon_2>0$ et $\varepsilon_2<0$.

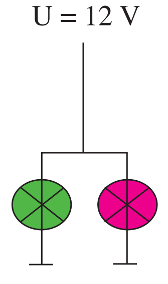
2) Quel est le rôle des résistances R4 et R5 ? Calculer leur valeur sachant que le courant maximal admissible par les diodes est Imax = 10 mA.

On choisit la valeur de R1 telle que : V1 = u_{C1}. On a alors : R1 = 5 kΩ.

1) Montrer que l'on a alors V2 = u_{C2}.
2) a) Déterminer V_{S1} lorsque : P < Pinf
Pinf < P < Psup
P > Psup
b) Déterminer V_{S2} lorsque : P < Pinf
Pinf < P < Psup
P > Psup
c) En déduire, dans chaque cas, l'état des diodes électroluminescentes.
d) Pour une pression de gonflage des pneus de 230 kPa, préciser l'état des diodes électroluminescentes.

Exercice 4 : Contrôle permanent des feux
Le montage présenté par le schéma ci-contre est destiné au contrôle de l'état des feux. Il se présente par un signal lumineux émis par une diode électroluminescente (LED) placée sur le tableau de bord. Elle doit pouvoir informer le conducteur lorsque l'une des deux (ou les deux) lampes du feux rouge est défectueuse.





Pour cela, par rapport au montage traditionnel (schéma gauche), on introduit dans le circuit naturel des lampes une petite résistance R1 qui fait office de capteur d'intensité.



Pour tout le problème, on pose les hypothèses suivantes :

- le générateur de tension est parfait et de valeur U = 12 V
- quelle que soit la température, les résistances gardent une valeur constante
- l'amplificateur opérationnel (ALI) est parfait, et compte tenu de son alimentation, son état haut se situe à U = 12 V et son état bas à 0 V.

A- Fonctionnement en comparateur :

On négligera les intensités des courants qui circulent dans les résistances R_2 et R_4 par rapport à l'intensité du courant dans les lampes en série R_1.

1- Exprimer littéralement le potentiel Ve^- en fonction des grandeurs R_4 R_5 et U, puis, calculer la valeur de R_5 pour que ce potentiel de référence soit Ve^- = 8V.

2- Exprimer littéralement le potentiel Ve^+ en fonction des grandeurs R_1, R_2, R_3, U. et i_1 l'intensité alimentant les deux lampes.

Calculer alors ce potentiel pour les trois cas :

a) deux lampes en fonctionnement i_1 = 3,50 A

b) une seule lampe en fonctionnement

c) deux lampes défectueuses.

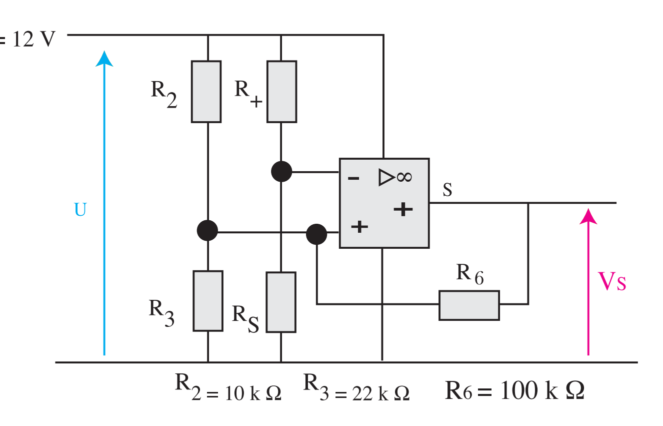
3- Des questions précédentes, déduire la valeur du potentiel de sortie Vs de l'amplificateur opérationnel et l'état de la diode pour chaque cas.

B- Fonctionnement avec une rétroaction positive :

Pour améliorer le montage précédent, on introduit une résistance R_6 entre la sortie et e^+.

Afin de faciliter l'étude, on analyse d'abord le modèle simplifié correspondant au schéma ci-contre. Le potentiel de référence reste fixé à Ve^- = 8,00 V.

Dans ce cas, le potentiel Ve^+ peut prendre les 2 valeurs suivantes :



$$Ve^+ = \frac{\frac{R_3 R_6}{R_3 + R_6}}{R_2 + \frac{R_3 R_6}{R_3 + R_6}} . U \qquad\qquad Ve^+ = \frac{\frac{R_3 R_6}{R_3 + R_6}}{R_2 + \frac{R_3 R_6}{R_3 + R_6}} . U$$

VE+ = U

1- A quel état de sortie de l'A.O correspond chacune de ces deux expressions ? Justifier.

2- Calculer numériquement le potentiel VE+ pour chaque état.



C- Comparaison de stabilité entre les deux montages

On admettra que le potentiel de référence reste constant VE+ = 8,00 V pour chaque montage. L'environnement électrique des lampes peut introduire des perturbations de 60 mV sur VE+.

1- Dans le premier montage, ce potentiel VE+ prend les valeurs de 7,85 V pour deux lampes en fonctionnement et 8,05 V pour une seule lampe. Quelles influences introduisent ces perturbations sur l'état de la DEL ?

2- Le deuxième montage (n°2 + rétroaction) offre maintenant des valeurs VE+ de 7,35 V pour deux lampes et 8,30 V pour une seule lampe. Analyser l'influence des perturbations sur l'état de la DEL et comparez au 1er montage.



Bibliographie

- **Circuits numériques " théories et applications "** : édition REYNALS GOULET 1996 de RNALD J TOCCI, traduction de André Lebel.
- **Electronique digitale** : édition DUNOD 1985, de PIERRE CANABIS avec la collaboration de Emmanuel Bernier.
- **Logique binaire** : édition MASSON 1984, de MICHEL AUMIAUX.
- **Cours d'électronique** tome1 : édition EYROLLES, de FRANCIS MILSANT.
- **Les microcontrôleurs PIC** : édition DUNOD, de CHRISTIAN TAVERNIER.
- **Applications industrielles des PIC** : édition DUNOD, de CHRISTIAN TAVERNIER.
- **Systèmes asservis " cours et problèmes "** : MICHEL LOBENBERG

Sites Web :

- http://www.tavernier-c.com/
- http://www.technologuepro.com
- http://www.datasheetcatalog.net/
- http://www.abcelectronique.com/
- http://www.motorola.com/
- http://www.fairchildsemi.com/
- http://www.abcelectronique.com/bigonoff/
- Sites divers et personnels.